Power-line Insulator Defect Detection and Classification

By

Usiholo Iruansi

213573619

A thesis submitted in fulfilment of the academic requirements for the Degree of Doctor of Philosophy in Computer Engineering in the School of Engineering

University of KwaZulu-Natal, Durban, South Africa

July 2017

UNIVERSITY OF KWAZULU-NATAL, COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE DECLARATION

The research described in this thesis was performed at the University of KwaZulu-Natal under the supervision of Prof. Jules-Raymond Tapamo and Prof. Innocent E. Davidson. I hereby declare that all materials incorporated in this thesis are my own original work except where acknowledgement is made by name or in the form of reference. The work contained herein has not been submitted in part or whole for a degree at any other university.

Signed:.....

Usiholo Iruansi

Date: July 2017

As the candidate's supervisor, I have approved/disapproved this thesis for submission.

Signed:.....

Prof. Jules-Raymond Tapamo

Date: July 2017

As the candidate's co-supervisor, I have approved/disapproved this thesis for submission.

Signed:.....

Prof. Innocent E. Davidson

Date: July 2017

UNIVERSITY OF KWAZULU-NATAL, COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE DECLARATION 1 -PLAGIARISM

1. The research reported in this thesis, except where otherwise indicated, is my original work.
2. This thesis has not been submitted for any degree or examination at any other University.
3. This thesis does not contain other person's data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This thesis does not contain any other person's writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
   (a) their words have been re-written, but the general information attributed to them has been referenced;
   (b) where their exact words have been used, their writing has been placed inside quotation marks, and referenced.
5. Where I have reproduced a publication of which I am an author, co-author or editor, I have indicated in detail which part of the publication was actually written by myself alone and have fully referenced such publications.
6. This thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis.

.

Signed:.....

UNIVERSITY OF KWAZULU-NATAL, COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE DECLARATION 2 -PUBLICATIONS

I, Usiholo Iruansi, declare that the following publications came out of this dissertation

1. U Iruansi, J R Tapamo, I E Davidson and M Khan, "Insulator Region of Interest using Active Contour Model and Speeded Up Robust Features". *In Proceedings of the* 2^{nd} *Eskom Power Plant Engineering Institute (EPPEI) Student Workshop, Eskom Academy of Learning*, Midrand, South Africa, June 8-9, 2015.
2. U Iruansi, J R Tapamo and I E Davidson, "An Active Contour Approach to Insulator Segmentation". *In Proceeding of the* 12^{th} *IEEE AFRICON International Conference*, Addis Ababa, Ethiopia, September 14-16, 2015, pp. 556-560.
3. U Iruansi, J R Tapamo and I E Davidson, "An Active Contour Approach to Water Droplets Segmentation from Insulators". *In Proceedings of the IEEE International Conference on Industrial Technology (ICIT2016)*, March 14-17, 2016, Taipei, Taiwan, pp.737-741.
4. U Iruansi, J R Tapamo, I E Davidson and M Khan, "Polymeric Insulator Condition Analysis Based on Hydrophobicity". *In Proceedings of the* 3^{rd} *Eskom Power Plant Engineering Institute (EPPEI) Student Workshop, Eskom Academy of Learning*, Midrand, South Africa, July 11-12, 2016.
5. U Iruansi, J R Tapamo and I E Davidson, "Insulator Condition using Local Binary Patterns Combined with Support Vector Machines". Manuscript accepted in 13^{th} *IEEE AFRICON International Conference*, Cape Town, South Africa, September 18-20, 2017.

6. U Iruansi, J R Tapamo and I E Davidson, "Hydrophobicity Classification of Insulators Based on Geometric Parameters". Manuscript under review in *Journal of computers (JOC)(Taiwan).*
7. U Iruansi, J R Tapamo and I E Davidson, "Classification of Insulator Condition using Local Binary Patterns with Support Vector Machines". Manuscript submitted to *IAENG International Journal of Computer Science, (IJCS).*
8. U Iruansi, J R Tapamo and I E Davidson, "Power-line Insulator Condition Classification". Manuscript submitted to *International Journal of Engineering and Technology Innovation (IJETI).*

.

Signed:.....

Dedication

To God Almighty, my wife, my kids, my mum, my entire family and in remembrance of my late Dad, Dr. David Aweriale Iruansi.

Acknowledgements

Glory, honour, adoration and praises to God Almighty for His mercy and strength to complete this research programme.

I am sincerely grateful to my supervisor, Professor Jules-Raymond Tapamo and co-supervisor, Professor Innocent E. Davidson. Your fatherly role, guidance, encouragement and teaching has been useful towards the completion of my research programme. Thank you Sir.

Special thanks to Eskom for the financial support through Eskom Power Plant Engineering Institute (EPPEI) Specialization Centre in High Voltage (DC), in the School of Engineering, University of KwaZulu-Natal (UKZN), Durban. Thanks to the past and present Directors; Prof. I.E. Davidson and Prof. D. Dorrell respectively.

Thanks to my industrial mentor, Mohammed Khan, to all my friends, associates, students and staffs of UKZN and the research group of Computer Vision, Image Processing and Data Mining, for making my dream come through. Special thanks to Dr. Andrew Eloka-Eboka and Dr. Temitope Mapayi for proof-reading this thesis. May God bless and reward you all abundantly.

I appreciate my wonderful mother, my in-laws, family members and siblings for their prayers and encouragements towards the completion of this research programme.

Lastly, I wish to thank my loving and caring wife, who gave me her genuine support. She was always there, bearing all the emotional and physical stress created in my absence and I also appreciate your patience, endurance, love, care and for holding the fort throughout the period of this research programme. Also, my kids are deeply appreciated as they waited patiently for daddy to finish the programme.

Abstract

Faulty insulators may compromise the electrical and mechanical integrity of a power delivery system, leading to leakage currents flowing through line supports. This poses a risk to human safety and increases electrical losses and voltage drop in the power grid. Therefore, it is necessary to monitor and inspect insulators for damages that could be caused by degradation or any accident on the power system infrastructure. However, the traditional method of inspection is inadequate in meeting the growth and development of the present power grid, hence automated systems based on computer vision method are presently being explored as a means to solve this problem speedily, economically and accurately.

This thesis proposes a method to distinguish between defectuous and non-defectuous insulators from two approaches; structural inspection to detect broken parts and a study of hydrophobicity of insulators under wet conditions. For the structural inspection of insulators, an active contour model is used to segment the insulator from the image context, and thereafter the insulator region of interest is extracted. Then, different feature extraction methods such as local binary pattern, scale invariant feature transform and grey-level co-occurrence matrix are used to extract features from the extracted insulator region of interest image and then fed into classifiers, such as a support vector machine and K-nearest neighbour for insulator condition classification. For the hydrophobicity study of the insulator, an active contour model is used to segment water droplets on the insulator, and thereafter the geometrical characteristics of the water droplets are extracted. The extracted geometrical features are then fed into a classifier to assess the insulator condition based on the hydrophobicity levels.

Experiments performed in this research work show that the proposed methods outperformed some existing state-of-the-art methods.

Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

ACM	Active Contour Model
ACWE	Active Contour Without Edges
CV	Cross Validation
DoF	Degree of Freedom
DoG	Difference-of-Gaussian
DT	Decision Trees
EPRI	Electric Power Research Institute
EPS	Electric Power System
GAC	Geodesic Active Contour
HC	Hydrophobicity Classification
HSI	Hue, Saturation and Intensity
KNN	K-Nearest Neighbour
LBP	Local Binary Patterns
LoG	Laplacian-of-Gaussian
NN	Nearest Neighbour
RGB	Red, Green and Blue
ROI	Region of Interest
ROIE	Region of Interest Extraction
SIFT	Scale Invariant Feature Transform
STRI	Swedish Transmission Research Institute
SVM	Support Vector Machines
UAV	Unmanned Aerial Vehicle

Chapter 1

General Introduction

1.1 Introduction

The Electric Power System (EPS) plays a very important role around the world, such as feeding homes and all areas of an establishment with a constant supply of electricity. The EPS is an interconnected network that is used to transport power from suppliers to consumers. It is made up of generation centres, distribution stations and diverse infrastructures. The generating centre is the source that produces electrical power which is then send to the distribution station via the transmission lines and finally to the demand centre [11]. Some of the most important components of transmission lines are insulators, conductors, spacers, dampers and bushings. One major component that can cause a great damage to the EPS if malfunctioning is the insulator. A faulty insulator can cause a change of voltage and electric current [12], which may lead to a great impact on the transmission and distribution of electricity. It is therefore important to monitor the insulators' degradation or deterioration in order to prevent failure and reduce outages and maintain or increase the performance of the power grid. Figure 1.1 shows insulators that are used to mechanically attach conductors to transmission towers/pylons and Figure 1.2 shows an insulator.

Figure 1.1: Pylon Insulators are used to mechanically attach conductors to transmission towers/pylons

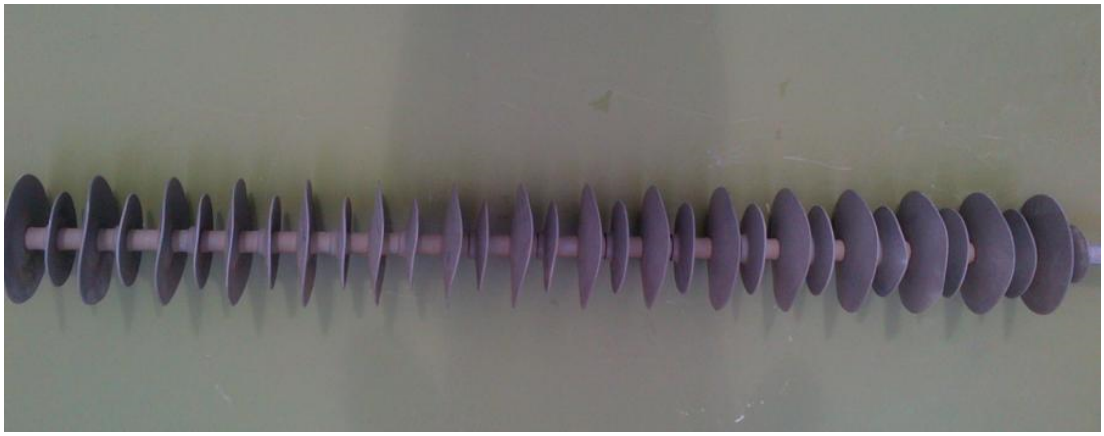


Figure 1.2: An Insulator

1.2 Problem Description and Motivation

A power-line insulator inspection is usually done manually (Figure 1.3), either directly on the lines or indirectly from the ground or air using a vehicle or helicopter respectively. When an inspection of a power-line insulator is done directly, it endangers the inspector since power-lines environment are polluted and dangerous with a high potential difference between the lines generating an electric and magnetic field in the vicinity of lines at normal condition and higher in the presence of defects [13]. Hence, it makes the on-site inspection of insulator condition using the traditional method very dangerous and unsafe to humans, but becomes an impractical task when monitoring and inspecting long lines spanning long distances with difficult terrain.

Figure 1.3: Climbing Inspector [1]

In aerial surveillance such as the helicopter assisted inspection (Figure 1.4), the helicopter is flown around the power-lines and a trained inspector uses a camera to track and film the power-line insulator for further analysis. This method is tedious, expensive and dangerous to the pilot and trained inspectors when the helicopter is too close to the power-lines. The constant vibration and translational movement of the helicopter can affect the sight control of the camera and as a result leads to image blurring [13], [14].

Figure 1.4: Helicopter inspection for insulators [2]

The advent of a digital camera system has reduced the number of inspectors for inspecting the insulators. Instead, inspectors are replaced by several cameras that are monitored by a single inspector through a computer aided system. This reduces the need for a large number of inspectors but is still prone to human error. This is because a single inspector needs to stay alert at all times in order to identify and track the insulator condition. However, the inspector can get tired and overlook some small defect on an insulator. This has necessitated having an automated system that can carefully detect the condition of insulators. Unmanned Aerial Vehicle (UAVs) (Figure 1.5) are also employed for insulator condition inspections. UAV employs the

Figure 1.5: Drone inspection for insulators [3]

Figure 1.6: Robot inspection for insulators [4]

principle of the helicopter because of its ability to fly around power-lines. Some of the problems with UAVs are similar to those of helicopter-assisted inspection, such as proximity and position control. An alternative approach to power-line inspection is a robot (Figure 1.6) which can move along the conductor, with its design capable of overcoming obstacles on the power-lines. The main advantage of the robot is its inspection accuracy due to close proximity to the power-lines and its low vibration increases the quality of image acquisition. The computer vision method has been identified as one of the robust solutions to identify and analyze insulators condition safely, speedily and accurately [15] and has been the subject of intense research in recent times. Image processing and computer vision methods are less expensive due to the advent of new technologies in the domain of digital imaging and the availability of low cost cameras. Therefore, incorporating computer vision into the robot gives a complete and robust automation system that can interpret the captured images.

1.3 Aims and Objectives

The main aim of this research is to design models for the classification of power-line insulator conditions. The specific objectives are:

- To develop an automated model for segmenting and extracting an insulator region of interest;
- To develop an automated model for the classification of insulator condition;
- To determine an optimized feature extraction model for insulator classification; and
- To develop an automated model for the hydrophobicity classification of insulators.

1.4 Contributions to Knowledge

The major contributions to knowledge of this thesis include:

- The development of an algorithm for the extraction of an insulator region of interest. The model uses the active contour method in combination with morphological operations to segment an insulator and further extract an insulator region

of interest. Extracting the region of interest helps optimize the classification of insulator condition (non-defectuous or defectuous).

- The design and implementation of a model for insulator condition. The model extracts an insulator region of interest and then uses feature extraction methods to extract features from the insulator image or series of insulator images while also attempting to reduce the extraction of features that are not stable or constant in an insulator image. These features are used to train a classifier and the classifier is used to determine the condition of any new insulator.

- The investigation of feature extraction method such as Grey Level Co-occurrence Matrix (GLCM), Local Binary Pattern (LBP) and Scale Invariant Features Transform (SIFT) in order to determine their suitability for the classification of insulator condition.

- The development of a model for insulator hydrophobicity classification based on geometric parameters. The proposed model uses a geodesic active contour approach in combination with morphological operation to segment water droplets on an insulator and then uses the geometric parameters to classify insulator condition.

1.5 Publications

1. U Iruansi, J R Tapamo, I E Davidson and M Khan, "Insulator Region of Interest using Active Contour Model and Speeded Up Robust Features". *In Proceedings of the* 2^{nd} *Eskom Power Plant Engineering Institute (EPPEI) Student Workshop, Eskom Academy of Learning*, Midrand, South Africa, June 8-9, 2015.

2. U Iruansi, J R Tapamo and I E Davidson, "An Active Contour Approach to Insulator Segmentation". *In Proceeding of the* 12^{th} *IEEE AFRICON International Conference*, Addis Ababa, Ethiopia, September 14-16, 2015, pp. 556-560.

3. U Iruansi, J R Tapamo and I E Davidson, "An Active Contour Approach to Water Droplets Segmentation from Insulators". *In Proceedings of the IEEE International Conference on Industrial Technology (ICIT2016)*, March 14-17, 2016, Taipei, Taiwan, pp.737-741.

4. U Iruansi, J R Tapamo, I E Davidson and M Khan, "Polymeric Insulator Condition Analysis Based on Hydrophobicity". *In Proceedings of the* 3^{rd} *Eskom Power Plant Engineering Institute (EPPEI) Student Workshop, Eskom Academy of Learning*, Midrand, South Africa, July 11-12, 2016.

5. U Iruansi, J R Tapamo and I E Davidson, "Insulator Condition using Local Binary Patterns Combined with Support Vector Machines". Manuscript accepted in 13^{th} *IEEE AFRICON International Conference*, Cape Town, South Africa, September 18-20, 2017.

6. U Iruansi, J R Tapamo and I E Davidson, "Hydrophobicity Classification of Insulators Based on Geometric Parameters". Manuscript under review in *Journal of computers (JOC)(Taiwan).*

7. U Iruansi, J R Tapamo and I E Davidson, "Classification of Insulator Condition using Local Binary Patterns with Support Vector Machines". Manuscript submitted to *IAENG International Journal of Computer Science, (IJCS)*

8. U Iruansi, J R Tapamo and I E Davidson, "Power-line Insulator Condition Classification". Manuscript submitted to *International Journal of Engineering and Technology Innovation (IJETI).*

1.6 Thesis Outline

This thesis contains seven chapters. The current chapter discusses the general introduction of the thesis, where the research context is given and problem description and motivation, aims and objectives, contribution to knowledge, thesis outline and conclusion are discussed. The rest of the thesis is organized as follows:

Chapter 2 describes the theory and background of the subject matter, such as the function of an insulator and its acquisitions, pre- and post-processing techniques, segmentation methods, feature extraction methods and classification. In addition, the advantages and disadvantages of these methods are highlighted and then reviews of previous researches in the related areas are discussed.

Chapter 3 discusses Active Contour Models (ACMs), such as the Active Contour Without Edges (ACWE) and Geodesic Active Contour (GAC). It proposes the implementation of ACM by Chan-Vese for insulator segmentation and the extraction of

insulator ROI. Also, comparison of ACWE and other segmentation method are carried out.

Chapter 4 introduces an automated method for the classification of power-line insulator condition using Local Binary Patterns and Support Vector Machines.

Chapter 5 describes an automated method for the classification of power-line insulator condition using Scale Invariant Feature Transform and K-Nearest Neighbour.

Chapter 6 proposes an automated method for the hydrophobicity classification of insulators using geometric parameter such as area and circularity.

Chapter 7 concludes the thesis and outlines future works.

1.7 Conclusion

This chapter has discussed the background of insulator condition monitoring, challenges faced, and motivation. It also highlighted the aims, objectives, contribution to knowledge and thesis outline.

Chapter 2

Insulator Condition based on Image Processing and Computer Vision

2.1 Introduction

This chapter presents an introduction to power-line insulators and its acquisition. It discusses the different stages involved in a complete design of understanding and analysis for insulator condition in the context of image processing and computer vision. The steps involved are image pre-processing, feature extraction and classification. In addition, related and previous research in insulator detection, insulator fault recognition and classification are discussed.

2.2 Insulators and their Acquisitions

Insulators are used to mechanically attach conductors to structures (such as pylons, towers, and poles); they provide an electrical insulation between an energized conductor and the grounded structure [16]. This makes the insulator an important component in the EPS, both in the distribution and transmission systems.

There are different types of insulating materials used for manufacturing insulators namely porcelain, glass and polymers; all of them possess the same characteristics of having a repetitive circular shape. Transmission line insulators can be categorized into

ceramic insulators and polymer insulators [16]. This thesis is based on polymer insulators. Polymer insulators may also be called polymeric, composite or non-ceramic insulators. They are characterized by being light-weight, compared to porcelain and glass insulator; the flexibility of its sheds minimizing breakage; the low cost of installation; less load to the supporting structure due to its light-weight, high-tensile strength compared to porcelain; high performance in polluted areas; and less cleaning due to its hydrophobicity. Hence, polymer insulators are most popular for outdoor high-voltage insulation [17], [18], [19].

A defect in an insulator defies its function, thereby causing great damage to the EPS. There are different types of insulator defects that are described by the Electric Power Research Institute [16], but this thesis focuses on insulator defect analyzed from two perspectives; physical defect such as split, torn or animal damaged as one set of defects and hydrophobicity as the other set of defects. Hydrophobicity (or hydrophobic) and hydrophicity (or hydrophilic) are terms used to describe the wetting properties of a surface by water [20]. A hydrophobic surface is water-repellent while a surface that attracts water is hydrophilic [21]. Hydrophobicity is not physically noticeable, but can be determined by hydrophobicity measure.

The technique for image acquisition in the first set (physical defect) is to use a digital camera to capture images of insulators. In the second set (hydrophobicity), insulator surfaces are sprayed with distilled water and digital camera is used to capture images of water droplets on the insulator surface from a fixed distance as water droplets approach a steady state.

2.3 Image Pre- and Post-processing

During image acquisition and after image analysis, noises may be introduced. These noises may affect the actual object of interest by creating unwanted effects such as unrealistic edges, lines, corners, artifact and blurring. To remove or minimize these effects pre- and post-processing need to be carried out. Pre-processing such as median filtering, Gaussian filtering and morphological operation, are mostly filtering operations. Post-processing at some stages of image analysis is needed to correct some anomalies introduced. The following post-processing methods are often used: dilation can be used to fill some gaps, erosion can be used to erode undesirable pixels, and opening

and closing are combinations of the two previous preprocessing operations. Before the pre-processing stage, captured images are usually converted from colour or RGB (colour Red, Green and Blue) format to other formats such as grey-scale or HSI (Hue Saturation and Intensity) which are usually convenient in most computer vision applications.

2.3.1 Median Filtering

Median filters are non-linear methods used in smoothing images while preserving edges or reducing edge blurring. It is used in image filtering and very useful for removing the salt and pepper kind of noise [22]. The median is computed by replacing the centre pixel of a nxn mask defined in an image with the median of all sorted pixel values of the $n \times n$ mask [23]. Figure 2.1 shows an example of the application of 3×3 median filter on a pixel.

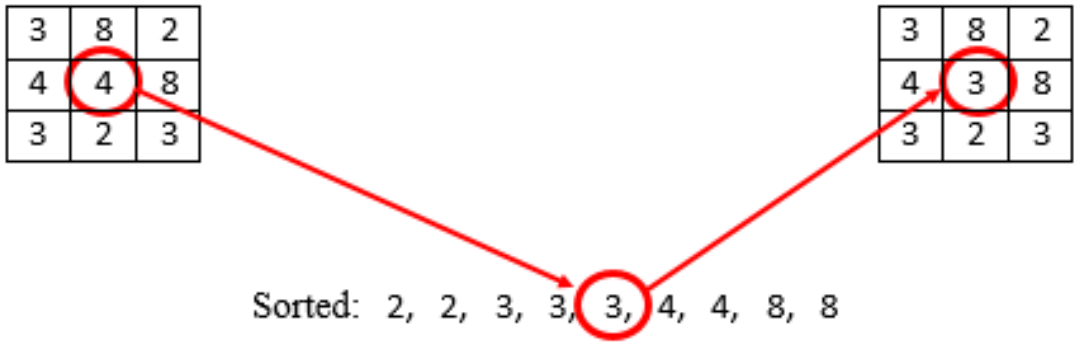


Figure 2.1: Application of the median filter

2.3.2 Gaussian Filtering

The Gaussian filter is a non-uniform low pass filter that is used in smoothing an image; it normally blurs the image. The operation of a Gaussian filter is to convolve a given image with a defined Gaussian kernel and the output is a smoothed version of the original image. The kernel is rotationally symmetric with no directional bias and its coefficients decrease with an increasing distance from the kernel center. Central pixels have a higher weighting than those at the edges. Gaussian filter has been used in [24] and [25] and it is stated in [25] that when an inappropriate variance is chosen, it has a significant impact on image smoothing, which may cause edge deletion.

2.3.3 Morphological Operation

Morphological operation is used in image processing to modify the spatial form or structure of objects within an image [26]. The main morphological operations are: erosion, dilation, opening, closing, top-hat and bottom-hat filters. Before any of these morphological operations are applied to a given image which may be either binary or grey-scale, it is necessary to define a structuring element. A structuring element is a shape or pattern which is used to interact with a given image, resulting in how the shape fits or misses the object in the image.

For morphological operations, dilation is when an object (foreground) in the image expands uniformly in spatial extent, whereas erosion is when a foreground in the image shrinks uniformly. Closing can be used to remove holes (background pixels) from the foreground and opening can be used to remove stray foreground pixels from image background. Top-hat and bottom-hat filters are good in correcting uneven illumination on a varying background. They are also used to extract small elements and details from a given image. Top-hat is the difference between the input image and its opening by a structuring element while the bottom-hat is the difference between the input image and the dilation with a structuring element. Top-hat preserves sharp peaks and improves contrast, while bottom-hat preserves sharp bottoms and improves contrast [27], [28], [29]. These morphological operations have been used in several studies, morphological pre-processing [30], image processing and quantitative shape analysis [31], background detection and illumination normalization [32], detection of cracked regions in ferrite [33] and image restoration [34].

A typical example of the implementation of morphological operation using the dilation operation is shown in Figure 2.2. The zero-values shown in Figure 2.2(a) represent the background and the one-values represent the foreground (the rectangular object in Figure 2.2(a)). Figure 2.2(b) is a structuring element with a three-pixel diagonal arrangement. The origin of the structuring element is represented with the rectangular border in Figure 2.2(b) which can be used to dilate the original image in Figure 2.2(a) by translating the structural element into several locations (at every one-value) in the original image. This increases the size of the object specified with the asterisk in Figure 2.2(c) and the final output in Figure 2.2(d).

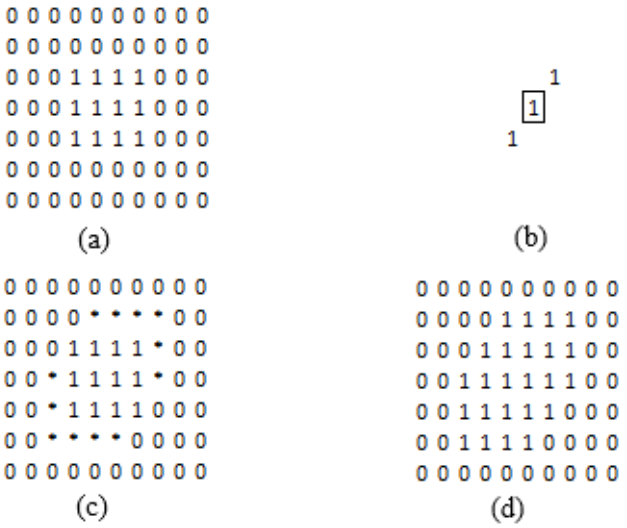


Figure 2.2: (a) Original image, (b) structuring element with three-pixel diagonal arrangement and the origin of the structuring element represented by a rectangular border; (c) structuring element translated into several locations on the original image and (d) final output. The zero-values and one-values represents background and foreground respectively.

2.4 Segmentation

Image segmentation is the process of partitioning an image into different sub-regions of foreground (object) and background based on desired features [27]. The segmentation helps to define the Region of Interest (ROI) of an object for further analysis. For example, to extract features from an insulator, it needs to be segmented in order to reveal the insulator ROI. There are different methods of segmentation, which can be categorized into edge-based and region-based segmentation depending on the implementation scheme.

2.4.1 Edge-based Segmentation

The Edge-based segmentation method identifies sudden changes in the intensity of an image, resulting in the connectivity of edges around the object of interest. Most edge-based techniques are based on the computation of a local derivative operator. The computation of the local derivative operation is by convolving an image with kernels such as Sobel, Prewith and Robert gradient. A critical disadvantage of the gradient operation is its sensitivity to noise. Therefore, a noise smoothing operation is required

before detecting edges in order not to include isolated pixels which may compromise the efficiency of edge detection [35]. A better option is convolving an image with Laplacian of Gaussian function which produces smooth edges as the Gaussian filter provides a smoothing effect [36]. The edge-based segmentation is extremely valuable in images with sharp edges or intensity transitions and generally low noise. The computational time is moderately lower than other segmentation techniques because the calculation depends on a local filtering operation.

2.4.1.1 Canny Operator

Canny operator [37] is computed by first smoothing the image using the Gaussian filter, then finding the intensity gradient of the smoothed image by computing the finite difference of the first order partial derivatives both in horizontal and vertical directions. Thereafter, the non-maximum suppression is applied in order to minimize th response to false edges. A double threshold is then applied to determine possible edges and finally, edge tracking by suppressing all the other edges that are weak and non-connected to strong edges.

The drawbacks of the canny operator is that the smoothing of the image may cause excessive smoothness leading to edge loss [38]. Furthermore, the dual thresholds are not self-adaptive, rather the values are set artificially. Therefore, threshold value set too high may lead to edge losses while threshold value that is too low may lead to excessive thickness of the edges. However, the canny operator is widely used in edge detection [39], [40].

2.4.1.2 Edge-based Active Contour

The first Active Contour Model (ACM) which is known as "snakes" was proposed by Kass et al. [41]. The principle of operation of the snakes is to use a gradient descent technique to deform a parametric curve drawn manually around the object in the image until the object region of interest is defined [42]. The snake has been used for object segmentation in [43], [44], [45] with promising results. The drawback of the snake method is that the curve is parameterized and cannot split and thereafter merge in order to identify more than one object in the image. Figure 2.3(a) shows an illustration with two circular objects. The red rectangle is the contour or curve and as

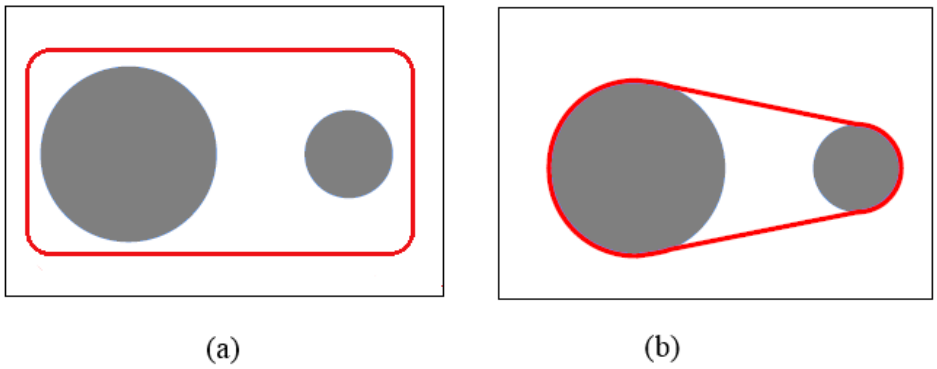


Figure 2.3: The inability for snakes to deal with changes in topology; (a) original curve, (b) final curve

the curve deforms to the boundary of the objects in Figure 2.3(b), it is not able to split in order to have two separate objects.

A further extension of the edge-based ACM is known as the Geodesic Active Contour (GAC). It was proposed by Caselles et al. [46], [47] in two separate studies. Caselles et al. [46], [47] introduced an additional term known as the stopping function, to the speed function. The GAC was the first level set implemented ACM for image segmentation. The level set theory for implementing active contours was first proposed by Osher and Sethian [48], and Sethian [49]. These researchers described a contour implicitly with a two-dimensional Lipschitz-continuous function characterized on the image plane. The Lipschitz-continuous function is known as the level set function. Other researchers however proposed a similar ACM in [50], [51], [52].

The active contour problem is proportionate to finding a path of minimal distance, called geodesic curve [36]. Advantages of the geometric approach over the parametric approach are that their representation does not require a choice of parameters and the topology of the curve is not fixed and can jump curves. This allows for the detection of any number of objects in the image, without knowing their correct number(s). This is accomplished with the presentation of a level set numerical algorithm for curve evolution, developed by Osher and Sethian [48]. The GAC has been used for object segmentation with promising results in [53], [54], [55], [56].

All ACM approaches that rely on edge function to stop contour evolution detect only objects characterized by gradient [7]. Therefore in practice, the discrete gradients

are bounded, the stopping function is never zero on the boundaries or edges of the object and the curve may pass through the boundary especially for the models in [46], [50], [57], [58]. The disadvantages of GAC may includes the fact that its output is affected in a noisy background in an image, that the application of a strong Gaussian smoothing to a noisy image also smooths the edges of the object and that GAC is relatively slow when compared to other conventional methods of segmentation [59].

2.4.2 Region-based Segmentation

Region-based segmentation utilizes information, such as colour, texture, intensity, or probability density function in guiding the motion of the curve [60], [61]. The advantage of the region-based segmentation over the edge-based approach is that it considers region areas rather than the local properties such as gradient. Region-based methods are thresholding and Active Contour Without Edges (ACWE).

2.4.2.1 Thresholding

Thresholding is the simplest method of segmentation. An intensity value known as the threshold is used to partition pixels in an image into foreground from its background based on their level of distribution. Thresholding produces a binary image from a grey scale image by making all pixels equal or below some threshold to zero and above to one. There are different types of thresholding techniques, namely, global or fixed thresholding, local or regional thresholding and adaptive or dynamic thresholding. In a global thresholding technique, a single threshold value is used which is a function of only the entire image [62]. In a local thresholding technique, the threshold value is a function of both the image and the local properties [62]. In an adaptive thresholding technique, a different threshold value is used for different regions in the image [63]. There are several ways of selecting threshold value; either manually or automatically by the use of a threshold algorithm. In the computation of the automatic threshold value, the mean or median is used but performs poorly in a noisy image [63]. A better technique is to create a histogram of the image pixel intensities and use the valley point as the threshold. However, the image histograms may not have well defined valley points which may lead to difficulty in threshold selection. Thresholding has been used for several object segmentations, medical imaging [62], insulator recognition [12], insulator hydrophobicity classification [10] and robust object segmentation [64].

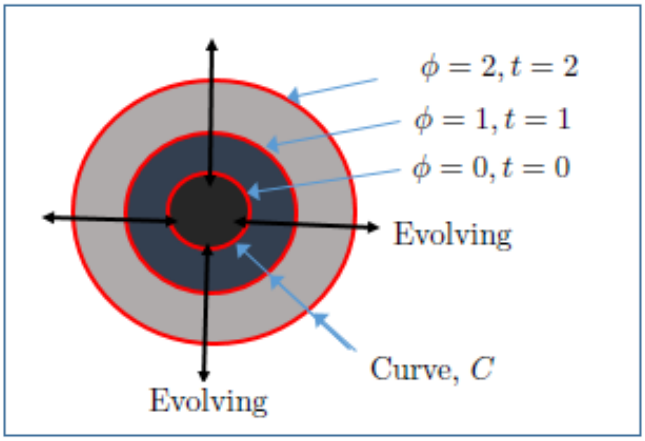


Figure 2.4: The level set function, ϕ evolution from its zero level set and curve, C propagation

2.4.2.2 Region-based Active Contour

The region-based method is strictly based on level set theory. In the implementation of the level set function on image segmentation, the zero level set is used to define the contour in order to have a border between a negative area and a positive area, so that the contour can be defined using the sign of the level set. As the level set function increases from its initial state, the contour propagates towards the inside or outside depending on the implementation (Figure 2.4). Thus, it can be stated that the level set function is equivalent to the evolution of the contour.

A different ACM without the utilization of an edge/stopping function to stop the contour/curve evolution was proposed by Chan and Vese [7]. It is a region-based method that is known as Active Contour Without Edges (ACWE) and it is based on Mumford-Shah [65] techniques. ACWE consists of the regularity part, which decides the smooth shape of contours and the energy minimization part, which searches for the consistency of a coveted feature inside a subset instead of searching edges during contour deformation. ACWE approximates the length of the curve by a function of the level set. This model relies on global properties such as intensities, texture and region areas, rather than considering the local properties such as gradient. The advantage of ACWE, is that it is able to segment an image that has smooth boundaries. This is possible because the evolution of the curve does not depend on gradient information,

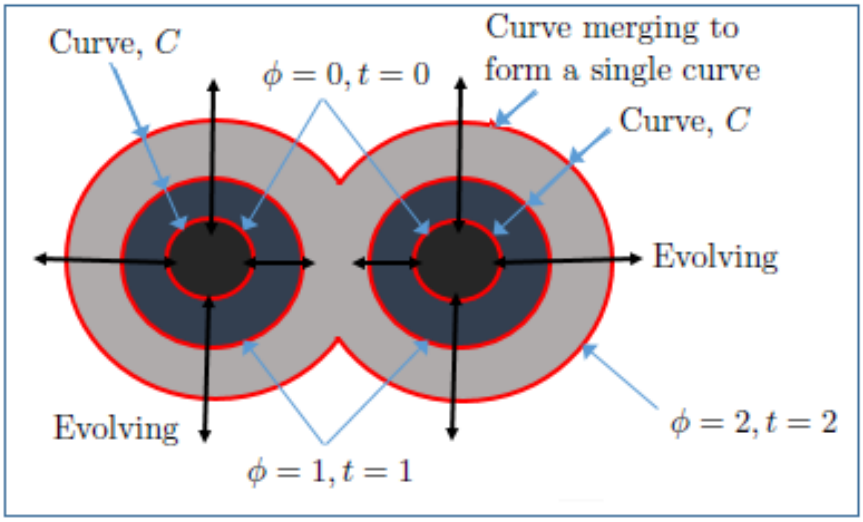


Figure 2.5: The level set function ϕ evolution from its zero level set with topology changes and curve propagation from initial C to a point of curve merging to form a single curve

therefore weak edges do not affect the final segmentation [66].

The advantage of the level set method is that contours can be separated into part or join together as the topology of the level set function changes (Figure 2.5). Therefore, more than one object can be detected in an image using level set methods and multiple initial contour can be used. One of the primary downsides of the level-set method is that it has high computational time due to the computation on an indistinguishable measurement from the image plane and furthermore on the grounds that partial differential equations have to be worked out [67]. High computational time can be minimized by utilizing more than one contour as initial contours for quick interaction with contour neighbours, thereby leading to fast convergence [36], [68]. However, the convergence speed of the level set approach is relatively slower than other methods of segmentation [36]. ACMs with level set technique have been used in various objects' segmentation [7], [35], [36], [60], [69], [70] [71], [72].

2.5 Feature Extraction

Feature extraction is very essential in image analysis and computer vision because it helps to describe relevant information contained in an object or pattern in order to

ease the task of object or pattern classification [73]. Feature extraction is also used for dimensionality reduction. When large amount of a data is being processed without much information, it is completely changed into a reduced arrangement of features also known as feature vector. The complete change of the input data into an arrangement of features is known as feature extraction. For well selected features (good features), a higher accuracy will be achieved for classification instead of using weak or poorly selected features (bad features) that will not properly describe the object and the use of the full size image. However, features can be described as information on region in an image that can be unambiguously recognized when seen in another image of the same scene possibly taken from a different view. There are different feature extraction methods, some of which are Grey Level Co-occurrence Matrix (GLCM), Local Binary Pattern (LBP) and Scale Invariant Features Transform (SIFT).

2.5.1 Grey Level Co-occurrence Matrix

GLCM [74], [75] is a statistical method proposed by Haralick for feature extraction. It is created by computing how regularly a pixel with an intensity value exists in a particular spatial relationship to another pixel with an intensity value in an image. The direction is often taken as $(0^o, 45^o, 90^o, 135^o)$. For example, considering Figure 2.6(a) as an image matrix and assuming that the position operator is one pixel on the horizontal to the right, the frequency of pair pixel is obtained by counting the number of times the pair pixel occurred in the original image. The pair pixels $(0, 0)$ in the red ellipse or circle in Figure 2.6(a) occurs twice and is represented as value 2 in the red circle in Figure 2.6(b). Similarly, the pair pixels $(2, 1)$ in the blue rectangle in Figure 2.6(a) occurs twice and represented as value 2 in the blue rectangle in Figure 2.6(b). This is applicable to all other pair pixels in Figure 2.6 in obtaining the output in Figure 2.6(b). However, the computed GLCM output will have a matrix that is equal to the number of distinct intensity values in the original image which is four (0, 1, 2, 3). From Figure 2.6(b), the co-occurrence matrix is computed. This is the division of all the considerable number of elements in the matrix by the sum of all elements in the same matrix.

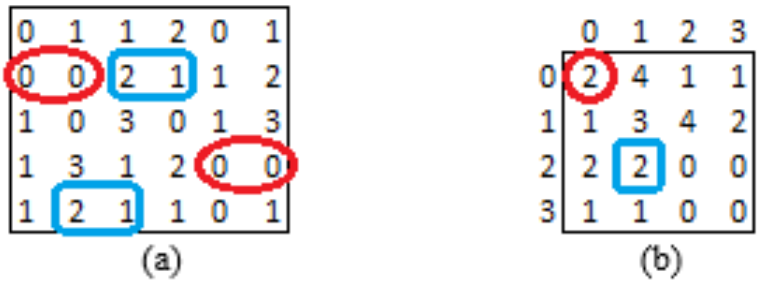


Figure 2.6: (a) Original image and (b) GLCM of original image in the horizontal direction (0^o) at a distance of one pixel

Once the GLCM is defined in a certain direction, it is normalized in order to ensure that the features are not influenced by the region's limitation. Fourteen GLCM features are also known as Haralick features and are: angular second moment (energy), contrast, sum of squares (variance), correlation, inverse difference moment (homogeneity), sum variance, sum average, entropy, sum entropy, difference variance, difference entropy, maximum correlation coefficient, information measure of correlation 1, and information measure of correlation 2. The Haralick features are defined in [74]. However, features with better performance are always selected from the fourteen Haralick features when carrying out an experiment. GLCMs have been used in different areas of image processing and computer vision namely, segmentation, object detection, recognition, fault detection, and classifications; insulator detection [8]; discriminate lungs abnormalities [76]; and brain tumour classification [77]. One main drawback of GLCM is that it is not rotation invariant.

2.5.2 Local Binary Pattern

LBP is a feature extraction method that is non-parametric. It is a grey-scale implementation, rotation invariant and defines the spatial structure of the local texture of an image [78]. In recent years, its application has increased in image processing and computer vision due to its computational simplicity and tolerance in monotonic illumination changes [79]. It has been used in numerous applications, such as defect detection [80], [81], visual inspection [82] and biometrics [83].

The principle of the original version of LBP [84] is used to create labels by using the centre value as a threshold for its neighbouring pixels and the results are binary numbers. For example, the value 4 inside the red circle in the image fragment in Figure 2.7 is used as the threshold for its neighbouring pixels, and the results is in binary which is read clockwise starting from the 0 inside the red circle. The resulting histogram is used as a texture descriptor. The restriction of the original version of LBP operator is that features computed in a local area of a 3×3 neighbourhood fail to capture the main features of some textures in large scale structures. As a result, it was modified to use neighbourhoods of various sizes [5].

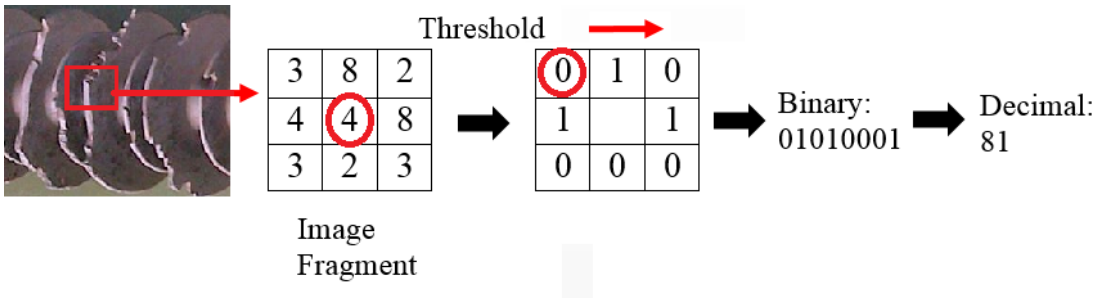


Figure 2.7: Basic LBP operator computation

The extended LBP uses circular neighbourhoods and bi-linearly interpolates values at non-integer pixels coordinates which allow for any radius and number of pixels in the neighbourhood. For example, in Figure 2.8(a), g_c represents the centre pixel with value 4 and the differences between g_0 and g_c are presented in Figure 2.8(b) in the same position of g_0, with value -1, likewise g_1 to g_7, and any value from the difference that is equal or greater than the value 0 is represented as 1 otherwise is 0 (Figure 2.8(c)). This gives a binary representation. In the circular neighbourhood, a local neighbourhood is referred to a set of sampling points P that are equally distributed on a circle of radius R which is focused at the pixels to be labelled, and the P that is not inside the pixels are inserted using bilinear interpolation, which considers any radius and any number of P in the neighbourhood. Figure 2.9 demonstrates a few cases of circular neighbourhood for different sampling points P and radius R defined as (P, R).

The LBP operator produces different output values that conform to different binary patterns produced from the neighbouring pixels. In the event that the image is

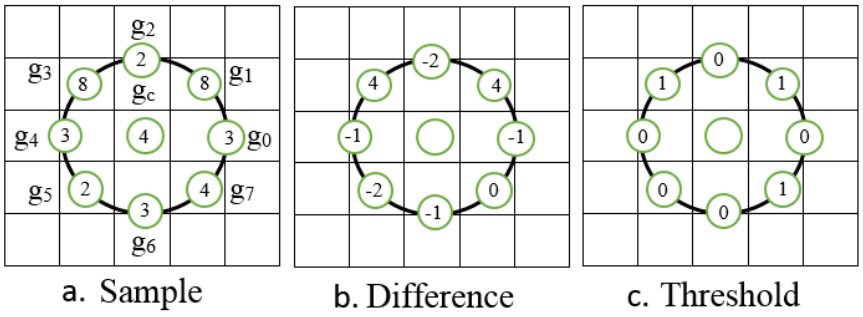


Figure 2.8: Circular neighbourhood LBP operator computation

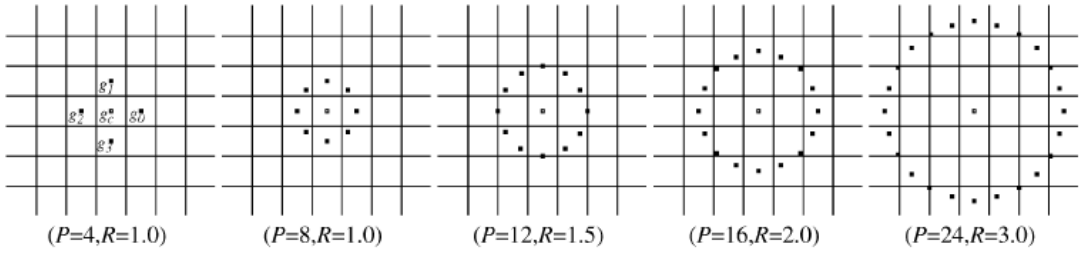


Figure 2.9: Different sampling points P and radius P, defined as (P, R) [5]

rotated, each closely encircling pixel in the neighbourhood will move accordingly along the border of the circle subsequently bringing about a different LBP value, except for patterns with $0s$ and $1s$ which also remain constant at all rotation angles. However, Ojala et al. [5] stated that the rotation-invariant LBP does not give very good discrimination because of the following two reasons namely, a great variation in the frequency of the 36 individual patterns and the unpolished quantization of the angular space at 45^o intervals.

A further extension to the original LBP operator uses a fundamental pattern called "Uniform" since they have a uniform circular structure that contains not very many spatial transitions [5]. Figure 2.10 describes the uniform pattern which functions as a template for micro-structures such as bright spot (0), edges of varying negative and positive curvature $(1-7)$ and flat area or dark spots (8). Figure 2.11 shows a non-uniform pattern. LBP is described as a uniform pattern if the binary pattern has at most two bitwise transitions from 0 to 1 or 1 to 0 when the bit pattern is traversed circularly (see Figure 2.10). For example, the patterns 00100000 and 00000001 have 2-transitions and 1-transition respectively for uniform patterns. This is because the patterns contain at

most two, 0-1 and 1-0 transitions. On the other hand, the pattern 10001000 is considered a non-uniform pattern since it has 3-transitions. This LBP operator proposed by Ojala et al. is a grey scale and rotation-invariant texture descriptor.

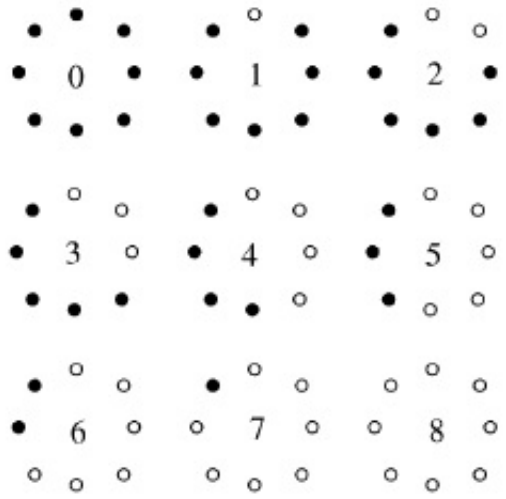


Figure 2.10: Uniform patterns with the number of transitions less or equal to 2 (≤ 2) and the numbers inside them correspond to their unique codes. The white and black circles denotes bit values of 1 and 0 in the 8-bit output of the operator [5].

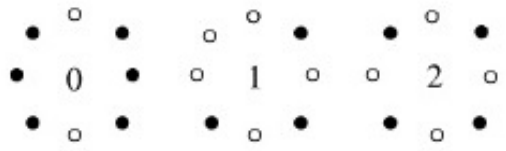


Figure 2.11: Non-uniform patterns with the number of transitions greater than 2 (> 2).

The use of uniform patterns has produced better recognition in many computer vision applications [85]. This is because research has shown that most natural images contains uniform patterns [5]. Also, there is reduction in the length of the feature vector and it is rotation invariant.

2.5.3 Scale Invariant Feature Transform

SIFT was proposed by David Lowe [6] and it is a popular algorithm for extracting features in images. The SIFT algorithm transforms an image into a set of feature vectors that are invariant to scale, rotation, affine distortion, changes in 3D viewpoint, noise and illumination changes. SIFT is used in many applications such as insulator recognition [86], [87], image classification [88], [89], [90] and facial recognition and detection [91], [92], [93], all with promising results. The main drawback of SIFT is that it is computationally expensive because of the high dimensionality of vectors (128-dimensional vectors) used to describe keypoints. The SIFT operation can be categorized into four stages which are used to identify image features. These four stages are, scale-space extrema detection, keypoint localization, orientation assignment and keypoint descriptor.

Scale-space extrema detection: The scale-space extrema detection is the first stage of the SIFT algorithm which searches an image over various scales and octaves in order to isolate extrema points of the image that are different from their neighbourhood. These extrema points are potential candidates for image features. SIFT is implemented by first rescaling the given image of each octave of scale-space by convolving more than once with a Gaussian convolution kernel into a set of scale-space images of lower resolutions (see images shown on the left of Figure 2.12) and then adjacent Gaussian images of the same scale are subtracted to produce DoG images (see the right part of the image in Figure 2.12). After each octave, the Gaussian image is down-sampled and the process repeated. Then to detect potential keypoints which are considered as the local maxima and minima in the DoG images that occurred on different scales (such as the one marked "X" in Figure 2.13), each pixel is compared with its eight (8) neighbours in the same or current DoG image and their nine (9) neighbours of the adjacent scales (marked with circles in Figure 2.13).

Keypoint localization: Once keypoints have been found, they are unstable which means that a lot of potential keypoints are generated with some along the edges, or have poor contrast (sensitive to noise). Thus, a refinement is required for more useful results. For discarding keypoints along the edges, the algorithm computes the principal curvatures from the Hessian matrix and it is used to threshold the points located along the edges, if the curvature that is perpendicular to the edge is large and the other along

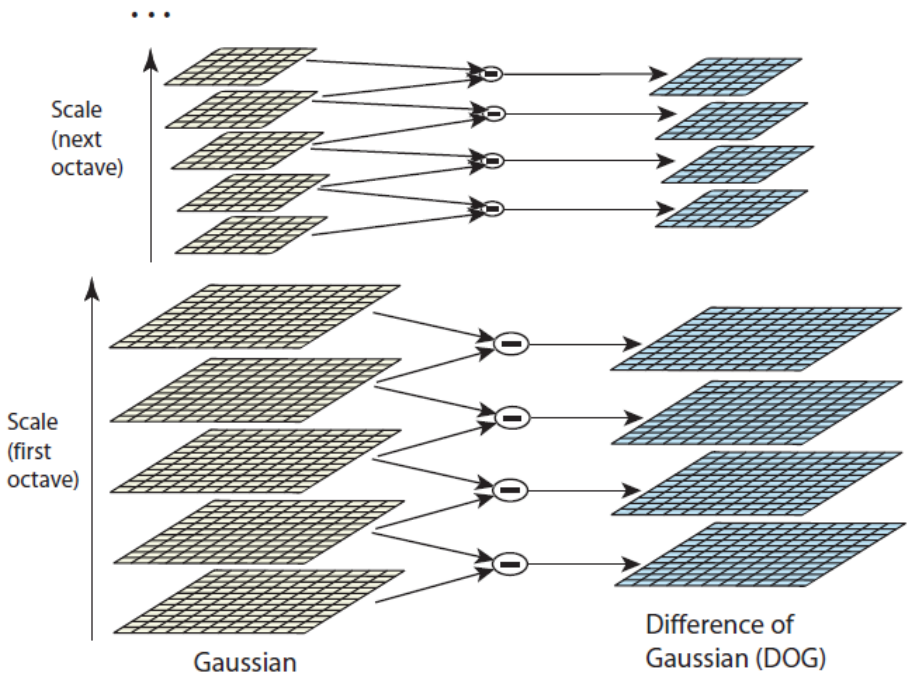


Figure 2.12: Rescaling the initial image of each octave of scale-space by repeatedly convolving with a Gaussian convolution kernel into a set of scale-space images (left images) and then adjacent Gaussian images of the same scale are subtracted to produce DoG images (right images). After each octave, the Gaussian image is down-sampled by a factor of s, and the process repeated [6].

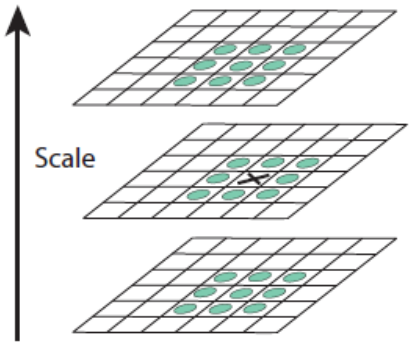


Figure 2.13: Potential keypoints which are considered as the local maxima and minima in the DoG images that occurred on different scales (marked "X"); each pixel is compared with its eight (8) neighbours in the same DoG image and their nine (9) neighbours of the adjacent scales (marked with circles) [6].

the border is small, the keypoint is discarded. If both curvatures are above a certain threshold, the keypoint is considered to be in a corner, and is kept. If both curvatures are small, the keypoint is located on a flat region and is also kept. For a poor contrast keypoint, if the magnitude of the intensity at the current pixel in the DoG image is less than a threshold, it is discarded. With this, a set of stable keypoints that are invariant to scale is achieved.

Orientation assignment: The next step is to allocate a standard orientation to the keypoints in view of the most dominant direction of the gradients in the local image properties. This step is important for achieving invariance to image rotation. In this process, the algorithm takes around each keypoint a neighbourhood whose size depends on the scale on which it was detected (higher scale, larger regions), and then calculates the magnitude and direction of the gradient at each pixel, to finally build an orientation histogram with 36 bins covering 360 degrees. The most prominent bin is taken as the gradient of the keypoint. For any other bin exceeding 80% of the most prominent bin, a new keypoint is created and its gradient is computed. For example, different keypoints can be created with the same location and scale, but with distinct orientations. At this point, it can be said that a set of keypoints invariant to scale with assigned orientation has been created.

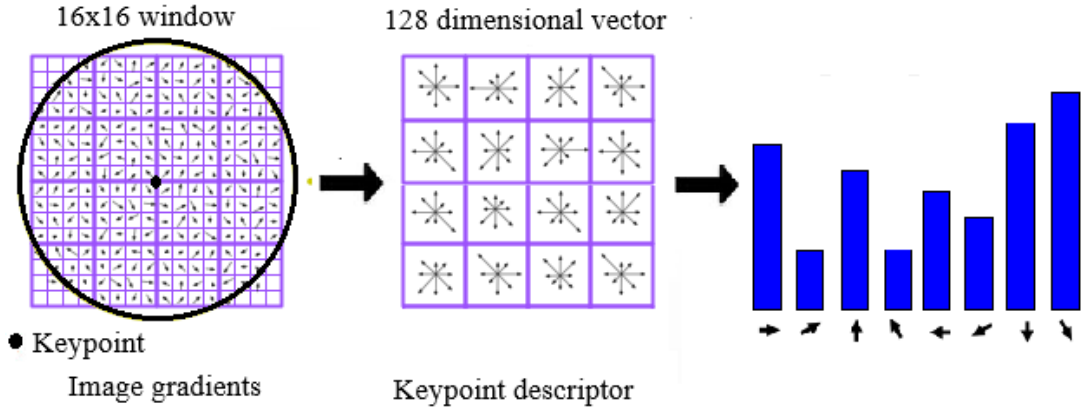


Figure 2.14: A keypoint descriptor creation

Keypoint descriptor: The last stage is to determine the keypoint descriptor, which is generated with $16 \times 8 = 128$ values from the SIFT feature vector forming the

keypoint descriptor that is invariant to scale, rotation and translations. Considering Figure 2.14 as an example, the left image of Figure 2.14 represents the keypoint descriptor generated by computing the gradient magnitude and orientation of every image sample point in an area around the keypoint location. These are then weighted by a Gaussian window, shown by the overlaid circle in Figure 2.14. The samples are then gathered into orientation histograms summarizing the contents over 4×4 sub-regions, as illustrated on the middle image of Figure 2.14, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region, as shown in the right side of Figure 2.14.

Finally, the feature vector is modified to reduce the effects of illumination change by normalizing the vector's length.

2.5.4 Geometric Parameters

The geometric parameters are the geometric features of an object. These features are numerical values computable from pixel or values in a region. The properties of these features are that they are easy to compute, and not affected by scaling, translation and rotation. An example of these features is the size of the object which is the total number of pixels in that object or region. The region in a binary image is equal to a 2D-distribution of foreground pixels within the discrete plane. Some of these geometric parameters are: perimeter, area, compactness, roundness or circularity, bounding box, convex hull and moments. These geometrical parameters have been used in [94] and [95] for determination of pharmaceutical ingredients.

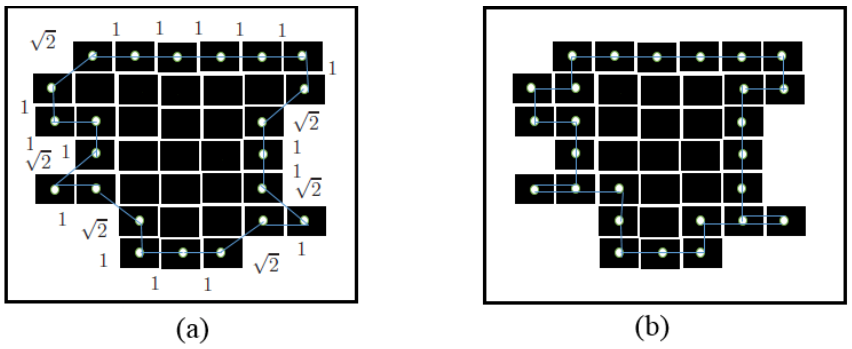


Figure 2.15: Computing the area and perimeter of an object

To obtain the area of an object, the pixels in the object are counted while the perimeter is the total length of the object boundary. For example, let the black rectangle boxes in Figure 2.15 represent pixels of an object, then the area is the total number of the rectangular boxes of the object which is 38 for both Figure 2.15(a) and Figure 2.15(b). For the perimeter computation, the horizontal and vertical movement is counted as 1 and the diagonal movement is counted as $\sqrt{2}$. Thus, the perimeter for Figure 2.15(a) is $(16 * 1 + 6 * \sqrt{2} \approx 24.49)$ and Figure 2.15(b) is 28. The area and the perimeter is the basis in computing other parameters.

For a particle that is irregularly shaped or non-spherical, the use of image analysis to determine the shape and size of an object is guaranteed [94]. However, more details of geometric parameters are discussed in Chapter **??**.

2.6 Classification

Image classification is a decision-making approach used to organize numerical properties of different image features into a few exclusive classes. The classes may be explicitly stated with a priori knowledge by a user (known as supervised classification) or automatically clustered (known as unsupervised classification) into a set of typical classes, where the user simply specifies the number of desired categories. The classification process involves training and testing phases. The training phase is the first phase of the classification process, where training classes are defined based on the description of the characteristic properties of image features. The testing phase is the second phase of the classification process and classifies image features based on feature space partitions. There are different classification techniques and some of which are Decision Trees (DT), Support Vector Machines (SVM) and K-Nearest Neighbour (KNN). These techniques are described in the following subsections.

2.6.1 Decision Tree

A decision tree (DT) is a machine learning method that is represented as a tree-like graph or model of decision-making and their possible outcomes [96]. The outcome of each test corresponds to a branch emanating from the node to another node representing a new test and recursively partitions the tree to the leaf node (final node), which provides the classification of the instance space [96], [97]. For example, Figure 2.16

shows a graphical representation of a basic DT classifier with three attributes (insulator present, insulator type and insulator disc). At the root node (left side of Figure 2.16), the tree may branch right or left relying upon whether the insulator is present or not (yes or no). The branches from each node may lead to a child node which may test the same attribute, or a different attribute or combination of attributes. The path from the root node to a leaf corresponds to a single classification rule. DT can also be linearized into decision rules [98], such as "*if* condition-1 and condition-2 *then* outcome" where the outcome is the contents of the leaf node, and the conditions along the path form a conjunction in the "if clause".

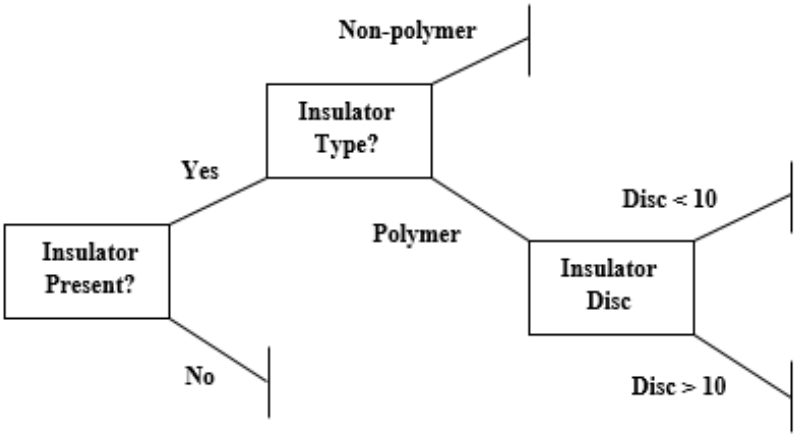


Figure 2.16: A simple decision tree classifier

DT has been used in [99], [100] with promising results. The advantages of DT are that they are easy to understand and interpret, possess value with little hard data, and allow the addition of a new possible sequence of events. The disadvantages are; mutual information from the DTs are biased in favour of those attributes with so many levels [101] and computation becomes difficult or complex particularly when there are uncertainties when many of the outcomes are linked.

2.6.2 K-Nearest Neighbour

The Nearest Neighbour (NN) algorithm for classification was developed and first introduced as a non-parametric method by Fix and Hodges [102] and has become known as K-Nearest Neighbour (KNN). This non-parametric classification is based on the "distance" between points or distributions. Later, Cover and Hart [103] introduced some of the formal properties of KNN which were investigated and established, such as new rejection approaches, refinements with respect to Bayes error rate, distance weighted approaches, and soft computing methods. In KNN classification, an object is classified into a class with a majority vote within the k-nearest neighbour [104]. For example, Figure 2.17 shows a set of samples divided into 2 classes. The test sample is represented as the black circle, that should be classified into the first class (blue rectangles) or to the second class (red triangles). For $k = 1$ represented as the solid line circle, the test sample (black circle) is classified into the first class (blue rectangles) because there are more rectangles inside the solid line circle than the triangles (rectangle = 3 and triangle = 2). For $k = 2$ represented as the dash line circle, the test sample is classified into the second class (red triangles) because there are more triangles inside the dash line circle than the rectangles (rectangle = 6 and triangle = 7). Note that k is an integer with a small value.

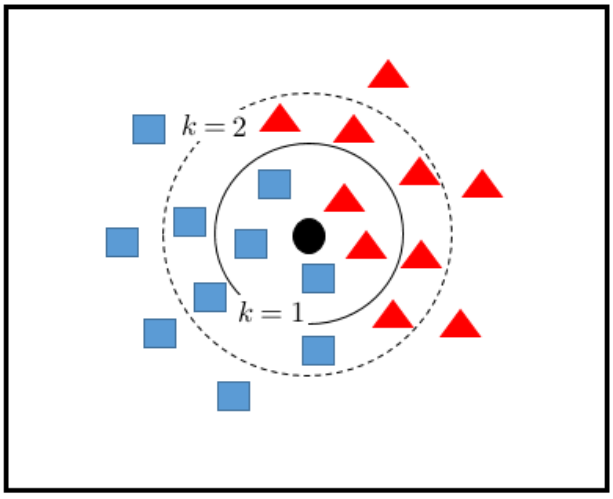


Figure 2.17: A KNN classification, with the test sample represented as the black circle, first class represented as the blue rectangles and second class represented as red triangles. $k = 1$ is the solid circle and $k = 2$ is the dash circle.

KNN has been applied in various domains such as recognition of insulators [86], recognition of faces [105], object tracking [106], prediction of economic events [104], forecasting the in-fluent characteristics of waste-water treatment plants [107] and classification for glasses [108] and hyperspectral image [109]. KNN performance is mainly dictated by the decision of k and the distance metric used [104], [110]. The advantages of KNN are its competitive classification performance in various domains, its robustness to high signal to noise ratio in the training set and its efficiency for a large training set [104]. Despite the advantages of the KNN classifier, it also has some disadvantages [111], [112], [113]: it has poor run-time performance for a large training set because of the computation of the distance of each query instance to all training samples and secondly, it is very sensitive to redundant or irrelevant features because all features contribute to the similarity and hence to the classification. However, careful feature selection or weighting can be used to avoid sensitivity to redundant features.

2.6.3 Support Vector Machines

SVM [114] is a supervised learning approach that has gained popularity due to its performance in the classification of noisy and high dimensional data. Some applications of SVM are in insulator condition analysis [115], insulator condition monitoring [116], the location of transmission line fault [117], fault diagnosis for power transmission system [118], [119], fault classification on transmission line [120], the diagnosis of skin illnesses [121], face recognition [122] and facial expression recognition [123]. SVM algorithm finds a decision boundary or a hyperplane that best classifies or separates data points over a vector space into different classes (see H_{pa} and H_{pb} in Figure 2.18). Considering Figure 2.18 as an example, let the rectangle data points belong to one class and the triangle data points belong to another class, the function of the SVM algorithm is to find a hyperplane H_{pa} or H_{pb} that best describes the separation of the two classes based on the maximum margin hyperplane. In Figure 2.18, both H_{pa} and H_{pb} separate data points. Dashed lines parallel to the solid lines indicate the amount one can move the choice limit (decision boundary) without causing misclassification of the data, but the distance (or margin) of hyperplanes H_{a1} and H_{a2} that pass through the nearest data point and parallel to the hyperplane H_{pa} is larger than the distance of hyperplanes H_{b1} and H_{b2} from H_{pb}. Therefore H_{pa} tends to perform better than H_{pb} for classification

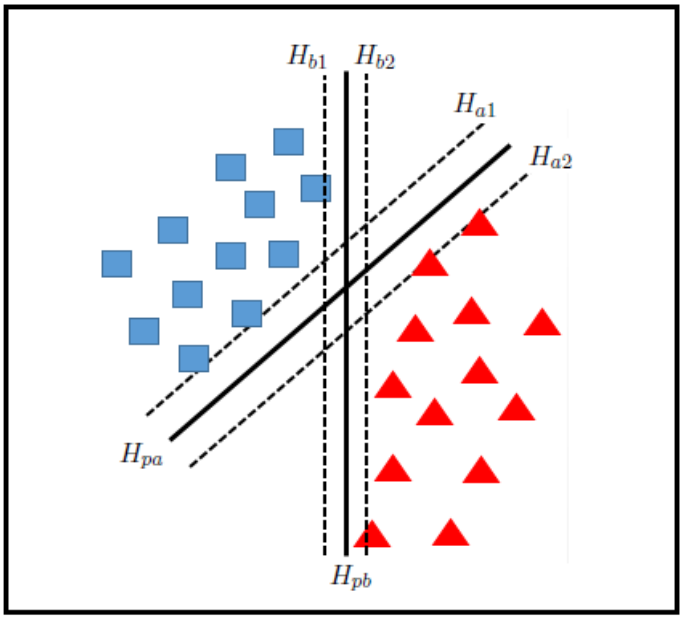


Figure 2.18: A linearly separable SVM classifier with the first class represented as the blue rectangles and second class represented as red triangles. H_{pa} or H_{pb} represented as the hyperplane or decision boundary, while H_{a1} and H_{a2} or H_{b1} and H_{b2} represented as the distance (margin) of the hyperplane.

of unknown samples due to its larger margin. Thus, the larger the distance or margin of the hyperplanes, the less the generalization error for the classification of unknown samples. Therefore, the conclusion can be drawn that H_{pa} performance is better than H_{pb}.

There are two types of SVMs namely, linearly separable SVM (Figure 2.18) and non-linearly separable SVM (Figure 2.19). For a complex dataset for which it is difficult to use linear SVM, non-linear SVMs are employed. The concept of non-linear SVM is to either introduce soft margin hyperplanes, or by transforming the original data vectors to a higher dimensional space where the data can be linearly separable. The main problem in the mapping of the data into higher dimensional space, is that the classifier becomes complex and the mapping function is not known. In order to overcome this problem, a concept known as the kernel trick is employed in the transformation process. There

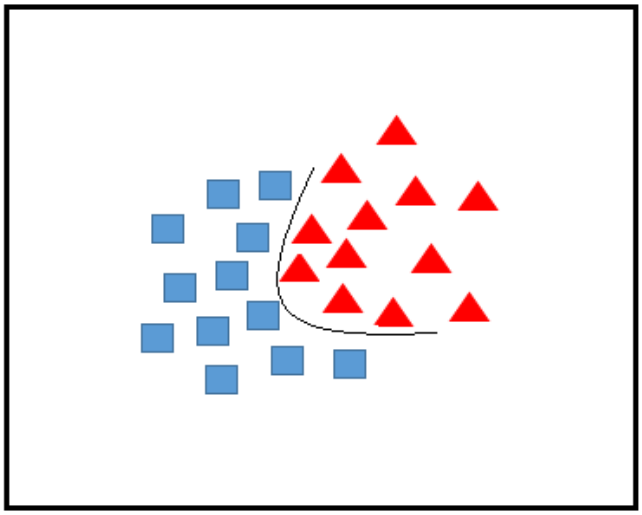


Figure 2.19: A non-linearly separable SVM classifier

are different kernels for SVM: linear, polynomial, radial basis function and sigmoid [124]. The main advantage of SVM over other classifiers is that the choice of the decision boundary is resolved only with the data points which have exactly the same distance from the decision plane [125]. These data points are known as the support vectors, which are the main compelling components in the training set in light of the fact that, if every single other data points are taken out, the algorithm will learn the same decision function [125]. This concept makes the SVM exceptionally one of a kind over other methods of classification that utilizes the entire data points in the training set for optimizing the decision function.

2.7 Previous Works

This section deals with the discussion of previous works relating to this research study. The first area describes the structural inspection to defects on insulators such as cracks and broken parts. The second area deals with hydrophobicity of an insulator under wet conditions. This defect can be identified by spraying distilled water on the insulator and then water droplets distribution and size are used as artefacts for insulator condition analysis. More importantly, there have been few research studies documented in the area of insulator condition based on image processing and computer vision.

2.7.1 Insulator Physical Conditions

Presently, there are research works on the inspection of power grids using infra-red and ultraviolet methods [8]. This study focuses on the use of texture features for insulator condition evaluation. There are few studies on the application of image processing and computer vision to insulator conditions (defect and non-defect). The last decade has necessitated the need for research incursion into the application of image processing and computer vision in various areas. Power-line insulator condition is an important area that comes to mind.

Gu and collaborators [126] proposed insulator fault detection. The fault detected is snow or ice coverage on an insulator. It is assumed that the fault is only accumulated on top or along the side of insulator caps. The fault is said to change the size of the disc edges and boundaries of the insulator which is said to be fixed and known. This is quite different from the fault proposed in this study which is based on cracks and broken parts of insulators.

Insulator fault detection were proposed by Mei et al. [127] and Ge et al. [128]. The fault detected is dirt on insulator surfaces. In [127], watershed algorithm is used for segmentation before the application of the HSI (hue, saturation and intensity) colour model. Also, in [128], the values of the R,G,B (red, green and blue) colour model were sampled and thereafter, judging based on fuzzy algorithm. The colour model is not a robust method for the detection of cracks and broken parts. Also, images were captured at a very close range showing just a single cap of an insulator, unlike the proposed images of this study showing the full insulator.

Li and co-workers [129] and Zhang et al. [130] computed edge descriptors on a dense grid. This creates a high number of false positives detections, which is not discriminatory enough when applied in a cluttered environment.

A simple method is proposed by Xinye et al. [12] using colour thresholding in insulator recognition. This method is not robust because it needs a well adjusted threshold parameter which may limit the method. Xinye and collaborators further proposed the detection of missing caps in [8]. The method proposed an accurate binary segmentation of the insulator using colour thresholding, then texture feature sequence extraction using GLCM, thereafter there was a further splitting of the insulator into ten parts. The method is limited by the choice of the threshold and the static partitioning

does not incorporate differently sized insulators or partially visible insulators. GLCM is also not rotation invariant and needs just one direction parameter during computation. Therefore, Xinye and collaborators applied Hough transform for insulator tilt correction since tilt affects insulator defect detection and morphological operation for enhancing the grey level distribution.

Discrete wavelet transform was used for feature extraction with SVM and Hidden Markov model for classification by Murthy and collaborators in [115] and [131] respectively. Both methods presented promising results. The Hidden Markov model outperformed SVM. Then Reddy and co-workers in [116] and [15] used discrete orthogonal Stockwell transform for feature extraction and adaptive neuro-fuzzy inference system and SVM for classification respectively. Discrete orthogonal Stockwell transform combined with SVM outperformed discrete orthogonal Stockwell transform combined with adaptive neuro-fuzzy inference system. Also, discrete orthogonal Stockwell transform with SVM outperformed discrete wavelet transform combined with SVM. This is because the discrete orthogonal Stockwell transform has an advantage of maintaining the phase properties of Stockwell and Fourier transforms, retaining the ability to go back to the Fourier domain.

Prasad and Rao [14] used a combination of LBP with histogram Fourier as an innovative approach to extract the rotation-invariant image feature which is computed by obtaining the discrete Fourier transforms of LBP histograms [132]. The features obtained are fed to SVM for classification purpose. It was shown that the combination of the LBP with histogram Fourier and its extensions outperformed non-invariant and earlier versions of the rotation invariant LBP in the rotation-invariant texture classification [5].

In the analysis of the physical condition of an insulator, uniform LBP and SIFT were adopted for feature extraction and compared with the GLCM method proposed in [8]. Uniform LBP is robust to monotonic grey-scale changes caused from illumination and a very efficient texture operator [133]. With its computational simplicity and discriminative power, it has gained popularity in various applications (see [79] for various applications of LBP). Also, SIFT is rotation, scale and translational invariant which makes it very robust, more stable and less prone to noise. These feature extraction methods are used in combination with SVM and KNN classifiers.

2.7.2 Insulator Hydrophobicity

To our knowledge, there are few research investigations on the application of image processing and computer vision to the analysis of hydrophobicity of insulator surfaces. Berg et al. [134] proposed hydrophobicity classification method for water droplets on insulators. This method is based on a function known as average of normalized entropies and its variants. Average of normalized entropy is implemented by calculating and analysing the distribution of the gray-level differences of nearest-neighbouring pixels in the horizontal direction. Shannon information entropy of the distribution is then utilized to determine the hydrophobicity of the specimen. However, because different HC levels may be identified in different sections of the insulator, the monotonic function is weak in this case [135]. Chen et al. [136] also used an improved average of normalized entropies with three other algorithms namely, seed point statistics, frequent magnitude analysis and shape factor for estimating the level of hydrophobicity online. It is stated in [136] that the improved shape factor is the best among all the methods being compared. But the drawback in this method of estimation of hydrophobicity level is that an operator is required to capture images online.

Tokoro and co-workers [137] described the aging process of silicone rubber (polymer) insulators using the effect of the shape of water droplets, but no method of water droplet detection was mentioned. Li et al. [138] mentioned estimating HC levels using the area ratio of the water droplet to the whole image and shape factor. Good results were achieved, but detection of water droplets was not considered.

Qiuxia and collaborators [139] applied a fuzzy means clustering approach to segment water droplets in a grey-scale channel. Images used in this work were taken indoors with multiple lightings but since it is rarely possible to have such illumination in outdoor scenarios, the performance of the algorithm should be investigated with images acquired from field tests [135].

Fractal dimension as a parameter has also been used to quantify the textures of insulator surfaces for classification by Thomazini et al. [140], but the texture of the images was sprayed with a mixture of isopropyl and water in different proportions. Also, Jarrar et. al [141] prepared samples using a mixture of distilled water and surgical spirit (a composition of methyl alcohol and ethyl alcohol). In [141], several feature extraction and selection techniques were used to extract statistical and textural features. These

feature extraction techniques namely, grey-scale co-ocurrence matrices, wavelet transform, discrete cosine transform, contourlet transform, radon transform and stepwise regression, were used with various classifiers, k-nearest neighbour, neural networks and support vector machines. Good results were achieved, but with a higher recognition rate in the use of fused features selected by a stepwise regression in combination with a neural network classifier. In most of these research works, the droplets are obtained using distilled water. Khalayli et al. [142] also used grey-level co-occurence matrices for feature extraction after segmentation of the water droplet on insulator using Sobel edge detection filter and then classification with a polynomial network or classifier. Good results were achieved, but the Sobel method is not robust because it needs a well-adjusted threshold parameter which may limit this method. Also, samples of HC level were classified into three groups namely, good (HC1-HC2), fair (HC3-HC4) and bad (HC5-HC6).

However, the method proposed in this thesis is an automated method that will be able to detect water droplets on insulators and classify insulators based on the geometric parameters of the water droplets.

2.8 Conclusion

This chapter discussed insulator acquisition, introductory theory to different methods used in insulator pre- and post-processing, segmentation, region of interest extraction, feature extraction techniques and classification methods. Finally, previous techniques for insulator defect detection, classification and insulator hydrophobicity were explored and discussed.

Chapter 3

An Active Contour Approach to Insulator Segmentation and Region of Interest Extraction

3.1 Introduction

The principle goal of this chapter is to determine and extract the ROI of insulator for further analysis. Before insulator ROI extraction, the insulator needs to be segmented. An active contour approach is used for the segmentation of the insulator from the background. Performance of the active contour model is evaluated by comparing it with other methods of insulator segmentation.

3.2 Active Contour Model

There are two major mathematical approaches of implementing ACM: the snakes and level sets. The snakes explicitly move predefined snake points using an energy minimization scheme, while the level set approach move contour implicitly as a particular level of a function. There are two approaches to image segmentation using ACM: edge-based and region-based. Edge-based active contours utilize an edge detector, to find the boundaries of objects in an image based on image gradient and then pull the contours to the detected boundaries. Region-based active contours use the statistical information of image intensity within each subset as opposed to seeking geometrical boundaries [143].

3.2.1 Snakes

The basic snake model is a parameterised spline guided in the image by a number of forces to a desirable position. These forces consist of internal forces or the internal energy of the contour, external constraint forces and image forces. The internal forces enforce a piecewise smoothness constraint. The image forces guide the snake curve towards desirable locations (such as edges, lines and subjective contours). The external forces pull the snake close to the coveted local minimum. These forces are imposed on the curve in the form of an energy function that measures the fitness of the snake and the energy function is minimised for desirable snake position and behaviour. These forces can be derived from the user interface, automatic attentional mechanisms or high level interpretations [41].

Given the position of a snake or contour parametrically is $C(s) = (x(s), y(s))$ then, the energy function is defined as:

$$E_{snake} = \int_0^1 [E_{int}(C(s)) + E_{img}(C(s)) + E_{con}(C(s))]ds \tag{3.1}$$

where E_{int} represents the internal energy of the contour due to bending, E_{img} is the image forces and E_{con} is the external constraint forces.

The internal contour energy is derived as:

$$E_{int} = (\alpha(s)|C'(s)|^2 + \beta(s)|C''(s)|^2)/2 \tag{3.2}$$

where $\alpha(s)$ and $\beta(s)$ are the user defined weights for controlling the tension or stretch or continuity of the contour and the rigidity or curvature of the contour respectively.

The internal energy of the contour is made out of a first order term that is controlled by a weight $\alpha(s)$ and a second order term that is controlled by a weight $\beta(s)$. In practice, the use of a large weight $\alpha(s)$ for the continuity term imposes a penalty on the changes in distances between points in the contour, while the use of a large weight $\beta(s)$ for the smoothness term imposes a penalty on the oscillations in the contour and will make the contour behave like a thin plate [144].

General formulation of image energy for an image $I(x, y)$ with the presence of lines, edges and termination, is defined as :

$$E_{img} = w_l E_l + w_e E_e + w_t E_t \tag{3.3}$$

where w_l, w_e and w_t are the weights of the lines, edges and termination. With higher weights, these salient features will have a larger contribution to the image force [144].

The line energy E_l can be represented as image intensity $I(x, y)$ at a particular point and the sign of E_l determines whether the line will be attracted to either bright or dark lines. The edge energy E_e, can be computed by the application of an edge detection operator, such that $E_e = -|\nabla I(x, y)|^2$. In order to avoid local minimal, a scale space continuation is used and the energy function is defined as $E_e = -|\nabla(G_\alpha * I(x, y))|^2$, where G_α is the Gaussian smoothing filter with the standard deviation α. The termination energy E_t, is computed by considering a slightly smoothed image $S(x, y)$ with gradient angle $\theta = arctan(S(y)/S(x))$, unit vector along the gradient direction as $\hat{n}$ and a unit vector perpendicular to the gradient direction as $\hat{n}_\perp$. Hence, E_t is defined as:

$$E_t = \frac{\partial \theta}{\partial \hat{n}_\perp} = \frac{\partial^2 S / \partial^2 \hat{n}_\perp}{\partial S / \partial \hat{n}} = \frac{S_{yy} S_x^2 - 2 S_{xy} S_x S_y + S_{xx} S_y^2}{(1 + S_x^2 + S_y^2)^{3/2}} \tag{3.4}$$

In the classic snake and other ACMs, the edge energy (edge detector) is regularly utilized relying upon the gradient of the image $I(x, y)$ to stop the deforming or evolving contour on the boundary of the desired object, but the line energy can still find its application [28]. In some systems, the user is allowed to interactively guide the contour, both in the initial placement and also in their energy terms. Therefore, the constraint energy E_{con} is used to interactively guide the contour towards or far from the lines and edges.

3.2.2 Level Set Methods

Level set methods essentially find the shape without parameterizing the contour, so the contour description is represented implicitly rather than explicitly via a Lipschitz continuous function $\varphi(x, y)$: $\Omega \rightarrow \Re$ defined on an image plane, Ω. The level set function is denoted as $\varphi(x, y)$. The zero level of the level set function, $\varphi(x, y)$, is defined as the contour, C, such that;

$$C = \{(x, y) : \varphi(x, y) = 0\} \forall (x, y) \in \Omega \tag{3.5}$$

Formulation of contour evolution using the magnitude of the gradient was initially proposed by Osher and Sethian [48] and it is defined as:

$$\frac{\partial \varphi(x, y)}{\partial t} = |\nabla \varphi(x, y)|(v + \varepsilon \kappa(\varphi(x, y))) \tag{3.6}$$

where v represents the speed in normal direction that tends to attract the contour to the object, the ε controls the balance between the regularity and robustness of the contour evolution and $\kappa(.)$ is the mean curvature of the level set function $\varphi(x, y)$ derived as:

$$\kappa(\varphi(x, y)) = div\left(\frac{\nabla \varphi}{\|\nabla \varphi\|}\right) = \frac{\varphi_{yy}\varphi_x^2 - 2\varphi_{xy}\varphi_x\varphi_y + \varphi_{xx}\varphi_y^2}{(1 + \varphi_x^2 + \varphi_y^2)^{3/2}} \tag{3.7}$$

where φ_x and φ_{xx} represent the the first and second order partial derivatives of $\varphi(x, y)$ with respect to x, while φ_y and φ_{yy} denote first and second order partial derivatives of $\varphi(x, y)$ with respect to y. The curvature term κ controls the regularity of the contours in same way as the internal energy E_{int} in the classic snakes model.

3.2.2.1 Edge-based Active Contour Model

An extension of the edge based ACM was proposed by Caselles et al. [46] called the geometric active contour (GAC) model, by adding an additional term called the stopping function $\varrho(I(x, y))$ to the speed function in Equation 3.6 and is given as;

$$\begin{aligned} \frac{\partial \varphi(x, y)}{\partial t} &= \varrho(I(x, y))|\nabla \varphi(x, y)|\kappa + \varrho(I(x, y))v|\varphi(x, y)| \\ &= \varrho(I(x, y))(\kappa + v)|\nabla \varphi(x, y)| \end{aligned} \tag{3.8}$$

This was the first level set that implemented ACM for image segmentation. The main goal of the stopping function $\varrho(I(x, y))$, is to stop the contour when it gets to the object boundaries. The stopping function $\varrho(I(x, y))$ used in Caselles et al. [46] and Malladi et al. [57], [58] is defined as:

$$\varrho(I(x,y)) = \frac{1}{1+|\nabla(G_\alpha * I(x,y))|^n} \tag{3.9}$$

where $(G_\alpha * I(x,y))$ is the smoothed version of the image $I(x,y)$, and n is equal to 1 or 2 depending on the implementation. The contour moves in the normal direction with a speed of $\varrho(I(x,y))((\kappa\varphi(x,y)) + v)$ and therefore stops on the object boundary or edges, where $\varrho(.)$ vanishes.

The constant v, acts as a correction term that is chosen so that the quantity $(\kappa + v)$ remains a positive value. The constant v, is the force pushing the curve towards the object, when the curvature κ becomes negative or null and also maintains the contour evolution by minimizing the enclosed area of the contour. Another ACM on level set that uses the image gradient to stop the curve was proposed by Malladi et al. [50] and it is defined as:

$$\frac{\partial\varphi(x,y)}{\partial t} = \nabla\varphi(x,y)\bigg(- v + \frac{v}{(M1 - M2)}(|\nabla(G_\alpha * I(x,y))| - M2)\bigg) \tag{3.10}$$

where $M1$ and $M2$ are the maximum and minimum values of the magnitude of the image gradient $|\nabla(G_\alpha * I(x,y))|$ and v is a constant. The speed of the evolving contour becomes zero on the points with highest gradients and stops the contour evolution at the boundary defined by strong gradients.

3.2.2.2 Region-based Active Contour Model

A different ACM without the use of edge-function to stop the contour evolution is the Mumford-Shah [65] segmentation techniques. The Mumford-Shah model can detect contours both with or without gradient. For example, objects with very smooth edges or with discontinuous edges. In addition, the model is based on level set formulation and the interior contours are automatically detected. The formulation of the Active Contour Without Edges (ACWE) model [7] is related to the Mumford-Shah segmentation techniques and the ACWE model proposed to minimize the energy $E_{cv} = E_1 + E_2$, where E_1 and E_2 are defined as:

$$E_1 + E_2 = \int_{\Omega 1} |I(x,y) - h_1|^2 dxdy + \int_{\Omega 2} |I(x,y) - h_2|^2 dxdy \tag{3.11}$$

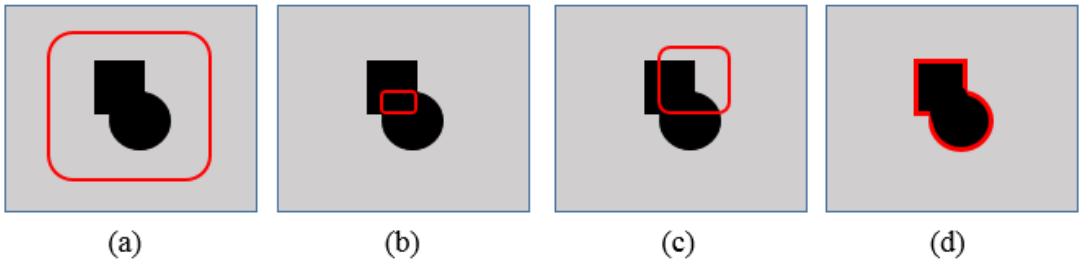


Figure 3.1: All possible cases in the position of the contour (a) $E_1 > 0$ and $E_2 \approx 0$, (b) $E_1 \approx 0$ and $E_2 > 0$, (c) $E_1 > 0$ and $E_2 > 0$ and (d) $E_1 \approx 0$ and $E_2 \approx 0$. The fitting term is minimized only when the curve is on the boundary of the object.

where $\Omega 1 = \{(x, y) : \varphi(x, y) > 0\}$ and $\Omega 2 = \{(x, y) : \varphi(x, y) < 0\}$, h_1 and h_2 are the average intensities inside and outside the evolving contour respectively, which are recalculated for each iteration of the contour evolution. The ACWE model has extra terms that minimized the length of the zero level set. The energy is at its minimum when the contour separates the foreground or object from the background. That is, if the contour is outside the object to be segmented, then $E_1 > 0$ and $E_2 \approx 0$, if the contour is inside the object to be segmented, then $E_1 \approx 0$ and $E_2 > 0$, if the contour is both inside and outside the object to be segmented, then $E_1 > 0$ and $E_2 > 0$ and only when the contour is on the object boundary that $E_1 \approx 0$ and $E_2 \approx 0$ (see Figure 3.1).

The ACWE model minimizes the fitting term and will also add some regularizing terms, such as the length of the contour C, and/or the area of the region inside the contour C. Therefore, the energy functional $E_{cv}(h_1, h_2, C)$ of the ACWE can be defined as:

$$E_{cv}(h_1, h_2, C) = \tag{3.12}$$
$$\mu \cdot L(C) + \nu \cdot A(\Omega 1) + \lambda_1 \int_{\Omega 1} |I(x, y) - h_1|^2 dxdy + \lambda_2 \int_{\Omega 2} |I(x, y) - h_2|^2 dxdy$$

where $L(C)$ is the length of the C, $A(\Omega 1)$ is the area of the region inside C, and ν, μ, λ_1, λ_2 are fixed parameters. ACWE Model in terms of level set function φ [7] is defined as:

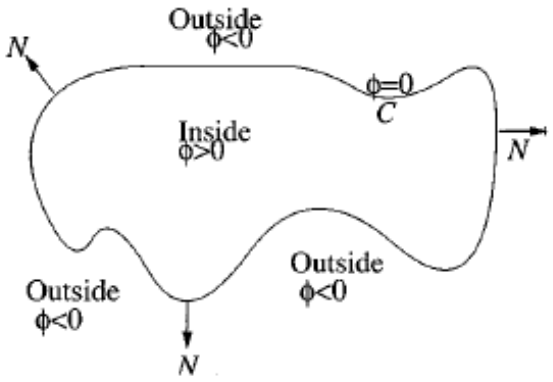


Figure 3.2: Contour or curve propagating in the normal direction with $\varphi > 0$ inside the contour, $\varphi < 0$ outside the contour and $\varphi = 0$ on the boundary of the contour [7]

$$E(h_1, h_2, \varphi) = \mu \int_\Omega \delta(\varphi(x,y))| \bigtriangledown \varphi(x,y)|dxdy + \nu \int_\Omega |H(\varphi(x,y))|dxdy \tag{3.13}$$
$$+\lambda_1 \int_\Omega (I(x,y) - h_1)^2 H(\varphi(x,y))dxdy$$
$$+\lambda_2 \int_\Omega (I(x,y) - h_2)^2 (1 - H(\varphi(x,y))dxdy$$

where h_1, h_2, H (Heaviside step function) and δ (one-dimensional Dirac measure) are defined as:

$$h_1(\varphi) = \frac{\int_\Omega I(x,y)H(\varphi(x,y))dxdy}{\int_\Omega H(\varphi(x,y))dxdy} \tag{3.14}$$

$$h_2(\varphi) = \frac{\int_\Omega I(x,y)(1 - H(\varphi(x,y)))dxdy}{\int_\Omega (1 - H(\varphi(x,y)))dxdy} \tag{3.15}$$

$$H(\varphi) = \begin{cases} 1, \text{ if } \varphi \geq 0 \\ 0, \text{ if } \varphi < 0 \end{cases} \tag{3.16}$$

$$\delta(\varphi) = \frac{d}{d\varphi} H(\varphi) \tag{3.17}$$

Thus, the gradient decent equation for the evolution of φ is derived as:

$$\frac{\partial \varphi}{\partial t} = \delta(\varphi)\left[\mu \nabla . \left(\frac{\nabla \varphi}{|\nabla \varphi|}\right) - \nu - \lambda_1 (I - h_1)^2 + \lambda_2 (I - h_2)^2\right] \tag{3.18}$$

where $\nabla.(\nabla\varphi/|\nabla\varphi|)$ is the curvature of the curve, that provides smoothing constraints during curve evolution thereby minimizing the total curvature of the contour.

3.3 Insulator Segmentation and Region of Interest Extraction

The insulator segmentation model presented in this thesis is based on active contour without edges proposed by Chan and Vese [7] and deals with balancing energy on both sides of the contour. It relies on global properties rather than taking into account the local properties. Due to the nature of the polymer insulator with rust, dirt, illumination and high signal to noise ratio, a preprocessing phase is required to improve the quality of images before the insulator segmentation and a post-processing phase is also required to further polish the final results [145].

3.3.1 Preprocessing

Captured images are normally in colour or RGB (Red, Green, Blue) format that are not suitable in some image applications such as insulator segmentation. However, a RGB image is converted into a HSI (Hue, Saturation and Intensity) image (Figure 3.3) using the following equations [27]:

$$H_H = \begin{cases} \alpha, & \text{if } B_B \leq G_G \\ 360 - \alpha, & \text{if } B_B > G_G \end{cases} \tag{3.19}$$

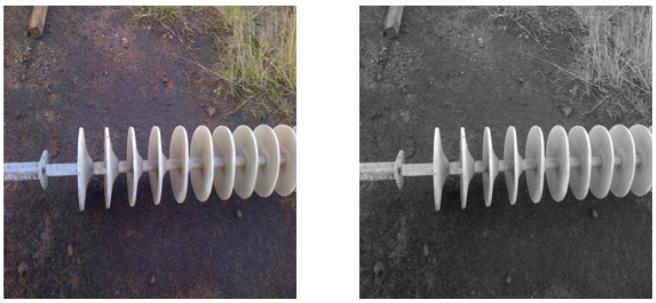
where H_H represents Hue, B_B represents Blue colour, G_G represents Green colour and α is derived as:

$$\alpha = cos^{-1}\left[\frac{\frac{1}{2}[(R_R - G_G) + (R_R - G_G)]}{[(R_R - G_G)^2 + (R_R - B_B)(G_G - B_B))]^{1/2}}\right] \tag{3.20}$$

where R_R which represents Red colour, saturation S_S and intensity I_I are given as:

$$S_S = 1 - \left[\frac{3[min(R_R, G_G, B_B)]}{R_R + G_G + B_B}\right. \tag{3.21}$$

$$I_I = \frac{1}{3}(R_R + G_G + B_B) \tag{3.22}$$



(a) (b)

Figure 3.3: Image conversion. (a) RGB image and (b) HSI image

3.3.2 Insulator Segmentation

Considering the properties of an insulator which has a rigid form with repetitive geometric structure and a distinctive circular shape of each cap, a region-based segmentation technique is required for insulator segmentation since uniformity of within sub-region is of main interest [146], [145]. Therefore, a region-based ACM known as ACWE based on Equation 3.13 is used for insulator segmentation. Fixed parameters are set to the following settings: $\mu = 1$, $\nu = 0$, $\lambda_1 = \lambda_2 = 1$ and then the mean intensities h_1 and h_2, and the Heaviside step function H, are defined using Equations 3.14, 3.15 and 3.16 respectively.

From Equation 3.13, the first term is the penalty on the total length of the edge contour for the segmentation, the second term is the penalty on the total area of the foreground (insulator) region found by the segmentation, the third term is proportional to the variance of the image grey levels in the foreground region and measures the uniformity of the region in terms of pixel intensity and the fourth term does the same for the background region. Minimizing the sum of the third and fourth terms leads to a segmentation into insulator (foreground) and background region that are uniform.

Thus, to segment insulator from the background image, Equation 3.13 is minimized with respect to h_1, h_2 and φ. With φ fixed, the average grey values of h_1 and h_2 are computed with Equations 3.14 and 3.15 respectively. Also, with h_1 and h_2 fixed, the evolution of φ is derived with Equation 3.18.

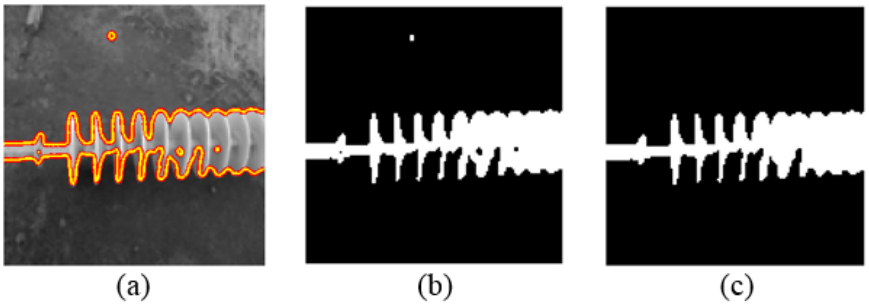


Figure 3.4: (a) Application of ACWE on Figure3.3(b), (b) Segmented image of (a) and (c) Post-processed image of (b) in order to remove the white spot on the background and the black spot inside the insulator in (b)

3.3.3 Region of Interest

The insulator region of interest (ROI) is extracted by first obtaining the coordinates and size of the foreground pixels (insulator) from the segmented image, then with the same size of segmented image and the original grey scale image, the coordinates and the segmented image size are mapped into the original grey scale image for placing a minimum bounding box over the entire insulator. Thereafter the insulator ROI is extracted.

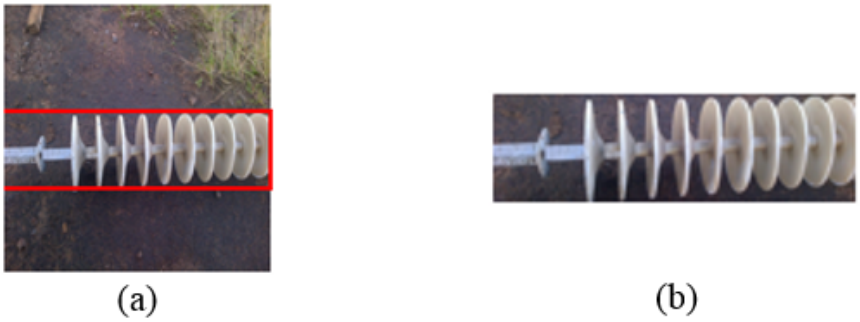


Figure 3.5: (a) Identification of insulator ROI and (b) Insulator ROI extraction

3.3.4 Algorithm for Insulator Segmentation and Region of Interest Extraction

Algorithm 1 presents the steps required for insulator segmentation and ROI extraction.

Algorithm 1 Insulator Segmentation and ROI extraction

Require: I ▷ Source Image
Ensure: E ▷ Extracted Insulator ROI Image
1: Convert the RGB image I into HSI and Save result as g
2: Initialize a level set function φ
3: **for** n = 1 to maximum value of n **do**
4: Compute $h_1(\varphi)$ and $h_1(\varphi)$ using equation 3.14 and 3.15 respectively
5: Compute φ^{n+1} by the discretization and linearization of equation 3.18
6: Check whether curve is stationary
7: **if** curve is stationary **then**
8: Escape from the for-loop
9: **end if**
10: **end for**
11: Save segmented image as sIm
12: BbIm = regionprops (sIm,"BoundingBox") ▷ Extract coordinates of the ROI from sIm using regionprops to compute bounding box index
13: lefttop = BbIm(1).BoundingBox(1)
14: rightbot = BbIm(1).BoundingBox(2)
15: width = BbIm(1).BoundingBox(3)
16: height = BbIm(1).BoundingBox(4)
17: rectangle ("position", [lefttop, rightbot, width, height]) ▷ Map the coordinates from sIm on g by defining a bounding box around the insulator
18: cropIm = imcrop(g,[lefttop, rightbot, width, height]) ▷ Extract insulator ROI from g
19: Save extracted insulator ROI as E
20: End

3.3.5 Experimental Results and Discussion

3.3.5.1 Dataset

To our knowledge, there is presently no publicly available dataset for insulator studies. Hence, the experiment and evaluation are based on the dataset of this research study. The dataset used contains insulator images on plain and fairly complex background. Twelve ground-truth insulator images were used for qualitative and quantitative performance analysis.

3.3.5.2 Evaluation

Figure 3.6 shows the results for the application of the proposed method. Figures 3.6(a),(d),(g) and (j) are the original images in RGB format, Figures 3.6(b),(e),(h) and (k) are the preprocessed images in the HSI format, Figures 3.6(c),(f),(i) and (l) show the results obtained after the application of ACWE and Figures 3.7(a),(d),(g) and (j) are the segmented images. Figures 3.7(b),(e),(h) and (k) are the post-processed images and Figures 3.7(c),(f),(i) and (l) are the ground-truth images used for evaluation.

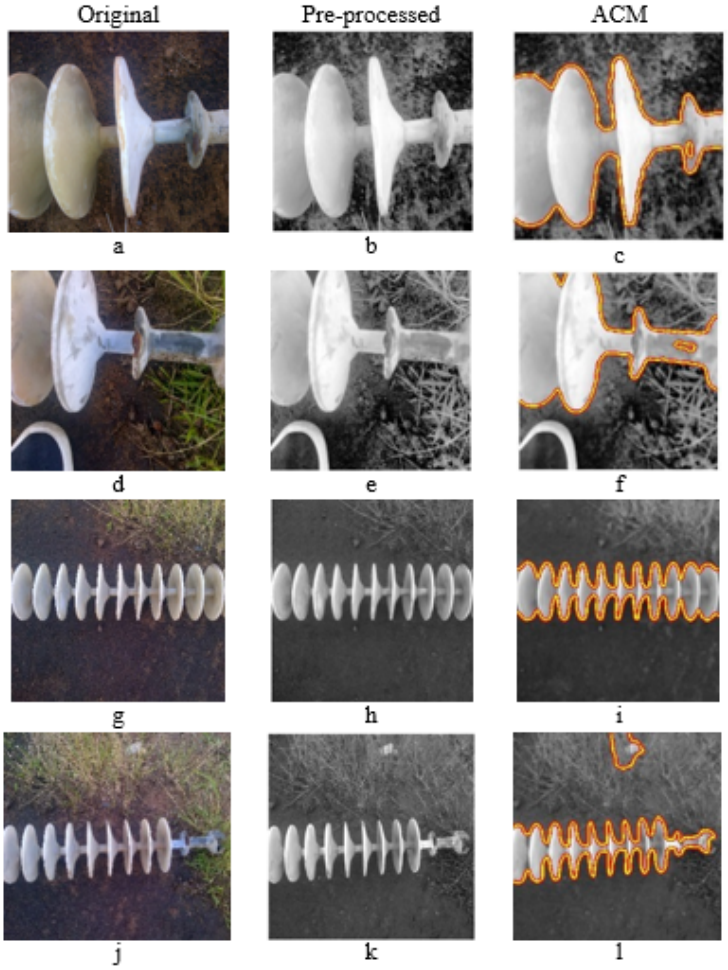


Figure 3.6: Insulation segmentation using the proposed method. Column one (a,d,g,j) represents the original images, column two (b,e,h,k) represents the pre-processed images and column three (c,f,i,l) represents the result of application of ACWE. Best viewed in colour

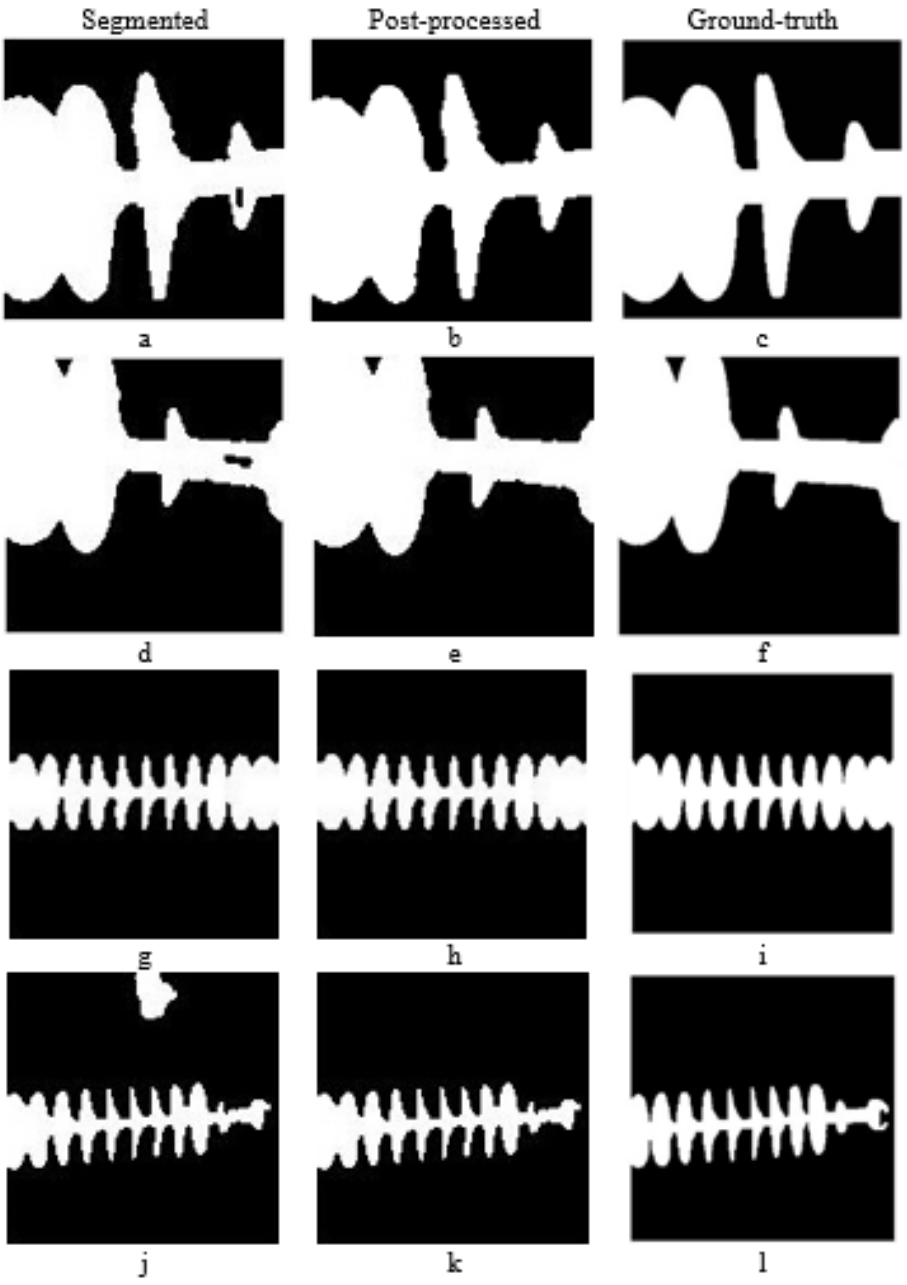


Figure 3.7: Insulator segmentation using proposed method and the ground-truth images. Column one (a,d,g,j) represents segmented images, column two (b,e,h,k) represents post-processed images and column three (c,f,i,l) represents ground-truth images

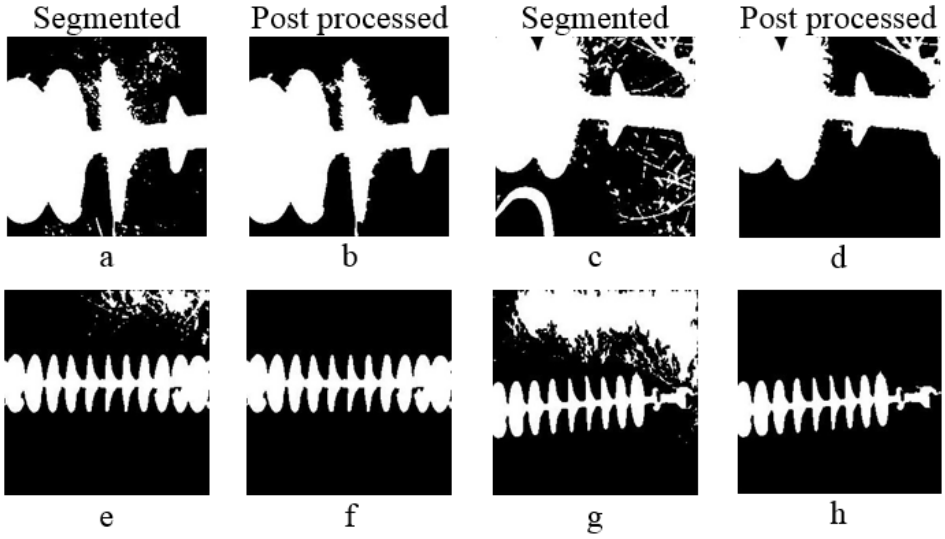


Figure 3.8: Insulator segmentation using Xinye's et al method. (a,c,e,g) represents segmented images and (b,d,f,h) represents the post-processed images of a,c,e,g respectively.

.

.

Also, Figure 3.8 shows the segmented images and post-processed images when the Xinye et al. [8] method is applied to Figures 3.6(a),(d),(g) and (j). The Xinye et al. [8] method for insulator segmentation is based on binary and morphology erosion, and thereafter the extraction of connected components and thresholding, which is based on the mean value.

Comparing the proposed method (Figure 3.7) to the Xinye et al. [8] method (Figure 3.8) qualitatively, it is obvious that the proposed method is more robust than Xinye et al.'s [8] method. However, images are post-processed in order to enhance the segmentation performance, such as in Figures 3.7(a) and (d) where background pixels are present in the foreground and in Figure 3.7(j), where foreground pixels are present in the background. It can be observed that the boundaries of insulators in Figure 3.8 are not as defined as the original image unlike the ACM approach with better boundaries. Also, the Xinye et al. [8] method has high signal to noise ratio after segmentation unlike the ACM approach. Also in comparing Figures 3.8 (a), (c), (e), and (g) to Figures 3.7 (a), (d), (g), and (j), there is a higher under-segmentation rate using the Xinye et al.

[8] method when compared to the proposed method.

Figure 3.9 shows the identification of insulator region of interest from the original images in Figure 3.6(a), (d), (g), and (j) using the proposed model. Figure 3.10 shows insulator ROI extraction. Both methods were able to identify and extract the ROI of the insulators. However, after the post-processing of the results of both methods on Figure 3.6(a), it is observed that the proposed method used a minimum bounding rectangle compared to the Xinye et al. [8] method in extracting the ROI of the insulator, which means that a good segmentation defines a better ROI.

Figure 3.9: Identification of ROI using proposed method. Best viewed in colour

Figure 3.10: Extraction of ROI from Figure 3.9

Figure 3.11: Identification of insulator ROI using Xinye et al. [8] method. Best viewed in colour

To further evaluate the effectiveness of the proposed model, the performance of the

proposed model of segmentation is quantitatively compared with the Xinye et al. [8] model based on the ground-truth of the insulators using different statistical criteria which are sensitivity Se, specificity Sp, precision Pr, accuracy Ac and F1-score Fs. These statistical criteria are defined as:

$$Se = \frac{TP_v}{TP_v + FN_v} \tag{3.23}$$

$$Sp = \frac{TN_v}{TN_v + FP_v} \tag{3.24}$$

$$Pr = \frac{TP_v}{TP_v + FP_v} \tag{3.25}$$

$$Ac = \frac{TP_v + TN_v}{TP_v + TN_v + FP_v + FN_v} \tag{3.26}$$

$$Fs = \frac{2TP_v}{2TP_v + FP_v + FN_v} \tag{3.27}$$

where the True Positive TP_v represents the total number of foreground (insulator) pixels in the segmented image that overlaps the foreground (insulator) pixels in the ground-truth. True Negative TN_v defines the total number of background pixels in the segmented image that overlaps background pixels in the ground-truth. False Positive FP_v represents the total number of foreground pixels in the segmented image that are background pixels in the ground-truth. False Negative FN_v are the total number of background pixels in the segmented image that are foreground pixels in the ground-truth.

In Table 3.1, it is observed that the proposed method has higher average performance rates in all statistical criteria over Xinye et al. method, denoting a better performance. Furthermore, the accuracy of the segmentation models are used to investigate the significance of the difference over both methods of segmentation. The mean and standard deviation of the accuracy for the statistical criteria are computed and

shown in Figure 3.12. The blue bars (standard error bars) at the middle of each bar plot (mean bars) represent the standard deviation. It is observed that both standard error bars overlap, which means that the difference between the two means of accuracy is not statistically significant.

Table 3.1: Performance analysis, with values of sensitivity, specificity, precision, accuracy and f1-score in percentage (%). P = Proposed method and X = Xinye et al. [8] method

	Sensitivity		**Specificity**		**Precision**		**Accuracy**		**F1-score**	
	P	X	P	X	P	X	P	X	P	X
Average	85.60	75.86	98.12	96.97	96.02	91.49	94.86	90.85	90.09	81.73

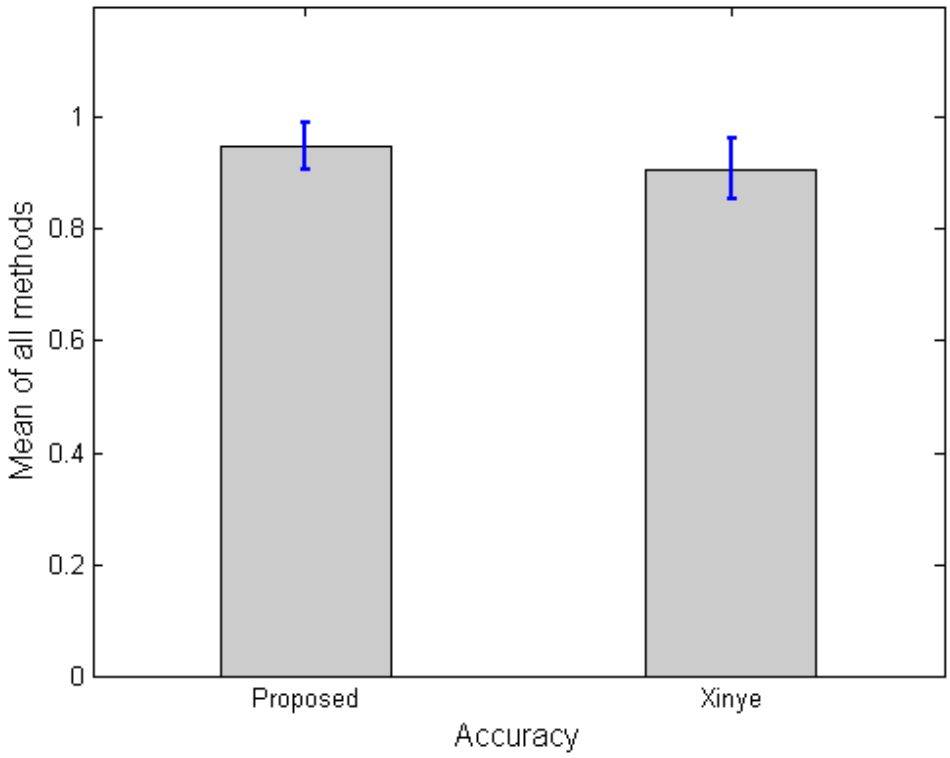


Figure 3.12: Mean and standard deviation of the proposed method of segmentation and Xinye et al. [8] method. Best viewed in colour

3.3.5.3 Test of Accuracy

To further investigate the significance difference of both segmentation methods, the t-test is employed. The t-test is a statistical hypothesis test used to confirm if there is a significant difference between two models Further, the t-test would be used to determine whether there is a significant difference between the mean accuracy value of the proposed method and the mean accuracy value of the Xinye et al. [8] method for insulator segmentation. The t-value formula is defined as:

$$t_{value} = |\bar{m}_1 - \bar{m}_2|(\sqrt{((Sd_1)^2/n_1) + ((Sd_2)^2/n_2)} \tag{3.28}$$

where $\bar{m}_1$ and $\bar{m}_2$ are the absolute mean values of the accuracy for the proposed method and Xinye et al. [8] method respectively, n_1 and n_2 are the numbers of insulator images for the proposed method and Xinye et al. [8] method respectively, Sd_1 and Sd_2 are the standard deviation of the accuracy for the proposed method and Xinye et al. [8] method respectively. The t-value parameters are presented in Table 3.2. Thus, substituting the values of the parameters in Table 3.2 into equation 3.28 gives the t-value ($t_{value} = 2.05$).

Table 3.2: t-value parameters for the dataset

Method	Number of Images	Mean	Standard Deviation
Proposed	12	94.86	4.12
Xinye et al. [8]	12	90.85	5.35

The degree of freedom (DoF) is computed as $(n_1 + n_2) - 2 = (12 + 12) - 2 = 22$. Generally in scientific analysis, to test for the statistical significance, a probability, P of $\alpha = 0.05$ is used. This probability value means that if the experiment is carried out 100 times, it means that 95% of the times, it will reject the null hypothesis H_o (where H_o means, there is no statistically significant difference between the proposed method and Xinye et al. [8] method) and 5% of the times, it will not reject the null hypothesis H_o. The critical value to determine H_o is to find the point at which the P value ($P = 0.05$) along the column of the t-table [147] for a two tail test that is independent (unpaired sample) intersect with the dof value (DoF = 22) along the row of the t-table [147]. Thus, a critical value of 2.074 is obtained. Since the t-value ($t_{value} = 2.05$) is not greater than the critical value 2.074, it means that the null hypothesis H_o is not

rejected, which indicates that there is no statistically significant difference between the proposed and Xinye et al. [8] methods. However, this actually depends on the size of the dataset used in an experiment.

3.3.6 Conclusion

In this chapter, insulator segmentation from plain and complex backgrounds using an active contour model and the extraction of insulator region of interest from the image context, has been presented. Also, it has been shown that the ACM method for insulator segmentation has well defined boundaries when compared to the Xinye et al. [8] method qualitatively. The proposed method has a higher accuracy over the Xinye et al. [8] method, but with the t-test analysis on accuracy, there is no statistical significant difference between the two methods. However, this actually depends on the size of the dataset used in an experiment.

Chapter 4

Classification of Insulator Condition using Local Binary Patterns with Support Vector Machines

4.1 Introduction

This chapter introduces an automated method for the classification of power-line insulator condition using Local Binary Patterns (LBP) with Support Vector Machines (SVM). The LBP operator is used to extract features from insulators and then fed into SVM to determine the condition of insulators. The traditional technique of insulator inspection is directly carried out on the power lines or indirectly from the ground or air, both of which put the life of the inspector at risk since the power-line environment constitutes high electric and magnetic fields. The manual technique is time consuming, expensive and becomes labour intensive in long distance with different and difficult terrains. The aim is to develop an automated model based on computer vision that will be able to identify and analyze insulators' condition safely, speedily and accurately. As a result, it will further minimize the failure caused by insulator defects which have major impacts on the transmission and distribution of electricity. Such impacts include: voltage drop, flow of leakage currents and losses to the power system [8], [146].

The flowchart of the proposed method for the classification of insulator condition

using LBP with SVM is shown in Figure 4.1.

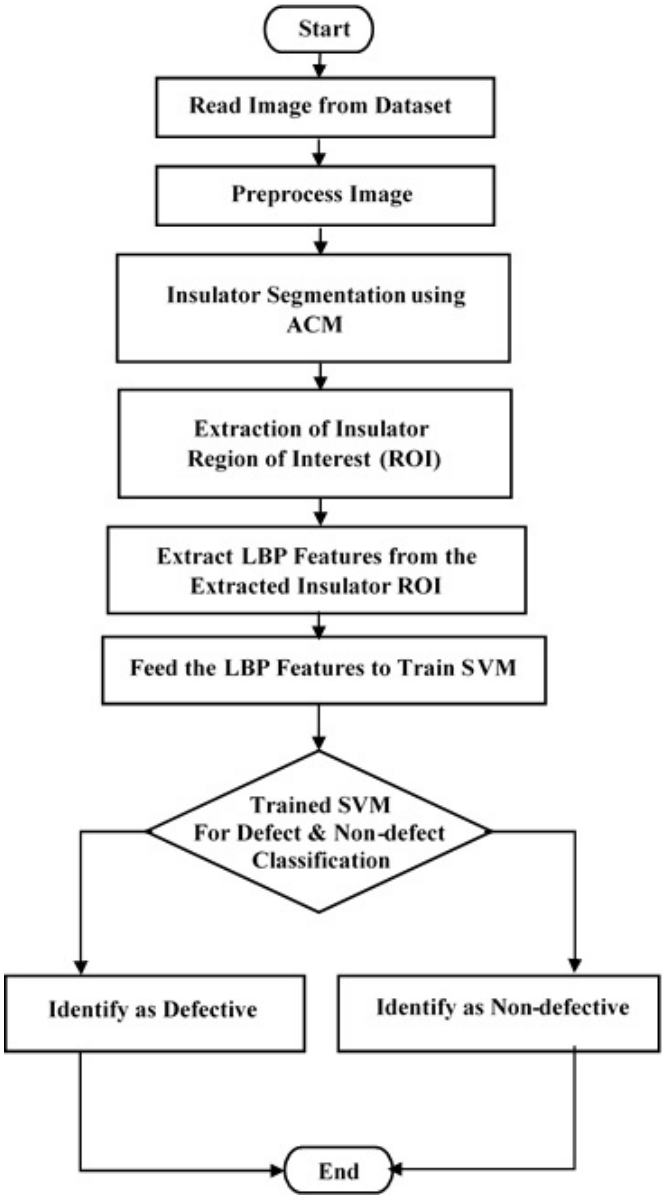


Figure 4.1: Flowchart of the proposed method using LBP with SVM

4.2 Proposed Method

4.2.1 Pre-processing

In order to reduce noise and improve the quality of images, pre-processing and post-processing steps are required. Colour images in Red, Green and Blue (RGB) format are converted into grey scale images (Figure 4.2). This conversion is done by forming a weighted sum of the R_R, G_G, and B_B components, such that the grey scale is derived as: $0.299 * R_R + 0.587 * G_G + 0.114 * B_B$. Then, morphological operation is applied to the grey scale images, for image enhancement.

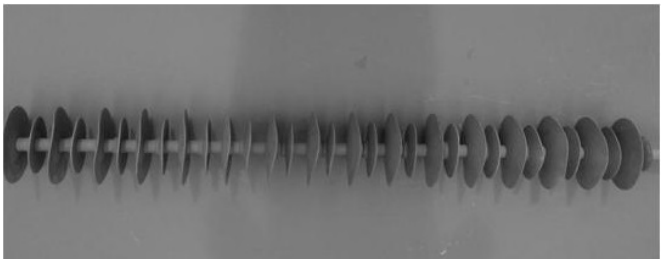


Figure 4.2: Greyscale image

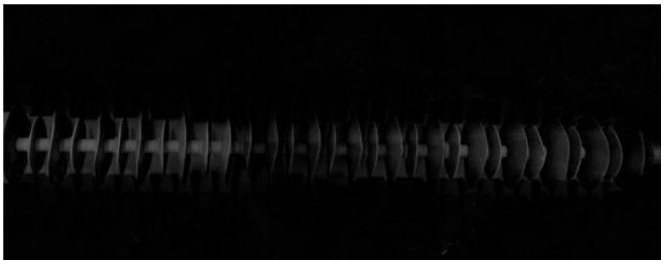


Figure 4.3: Top-hat filtering on grey scale image in (Figure 4.2)

Insulator images are enhanced using the top-hat filter [29] (Figure 4.3) which is very good in correcting uneven illumination on a varying background. It is used to extract small elements and details from the given insulator. It is defined as the difference between the input insulator image and its opening using a disk shaped structuring element. The top-hat is defined as:

$$T(I) = I - (I \; o \; S) \tag{4.1}$$

where I is the insulator image in grey-scale and S is the structuring element which is an open disk of radius 10. The opening operation o is the dilation of the erosion of a set M by a structuring element S, and the opening operation is defined as $[M\ o\ S\ =\ (M\ \ominus\ S)\ \oplus\ S]$. The symbols $\ominus$ and $\oplus$ denotes erosion and dilation respectively.

The grey scale image obtained is further enhanced by the application of a morphological operation [27] known as dilation (Figure 4.4). The dilation of grey scale images is the replacement of grey level values at any point with the maximum intensity value covered by the structuring element. This can be defined as:

$$(I_d\ \oplus\ S)(x,y) = max[I_d(x-x',y-y') - S(x',y')|(x',y') \in D_S \tag{4.2}$$

where I_d is the image and D_S is the domain of the structuring element S, and (x,y) are pixel coordinates.

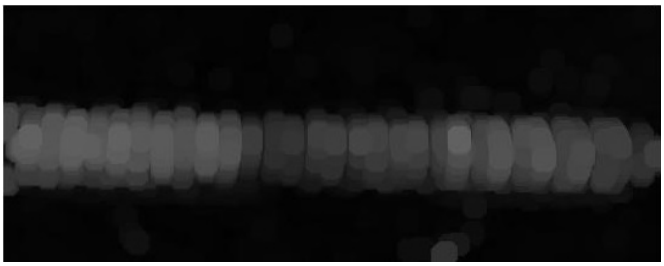


Figure 4.4: Grey-scale morphological dilation on insulator image in Figure 4.3

The output image of the dilated image is subtracted from the original grey scale insulator image for further analysis on insulator segmentation (Figure 4.5). The subtracted image is defined as:

$$I_s = I - I_d \tag{4.3}$$

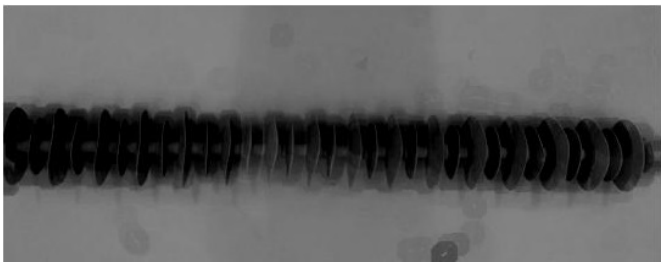


Figure 4.5: Subtraction of dilated image (Figure 4.4) from the original grey-scale image (Figure 4.2)

4.2.2 Insulator Segmentation

The region-based ACWE by Chan-Vese is used for insulator segmentation and it is based on Equation 3.13 in Chapter 3. The fixed parameters are set to the following settings: $\mu = 1$, $\nu = 0$, $\lambda_1 = \lambda_2 = 1$ and then the mean intensities h_1 and h_2, and the Heaviside step function H, are defined using Equations 3.14, 3.15 and 3.16 respectively.

Thus, to segment the insulator from an image, Equation 3.13 is minimized with respect to h_1, h_2 and φ. With φ fixed, the average grey values of h_1 and h_2 are computed with Equations 3.14 and 3.15 respectively. Also, with h_1 and h_2 fixed, the evolution of φ is derived using Equation 3.18.

Figure 4.6 shows the initial curve φ in an image I, where h_1 and h_2 are the average intensities values inside and outside φ respectively and the length of the edge contour is evolved in order to fit the boundary of the insulator such as in Figure 4.7. At this point, the curve becomes stationary, the interclass variance is minimized and the segmented image is achieved as shown in Figure 4.8.

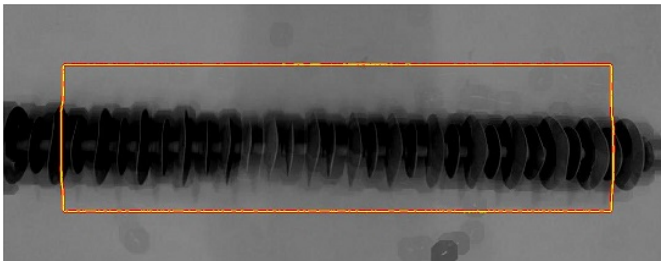


Figure 4.6: The initial curve C on the implementation of ACM on Figure 4.5. Best viewed in colour.

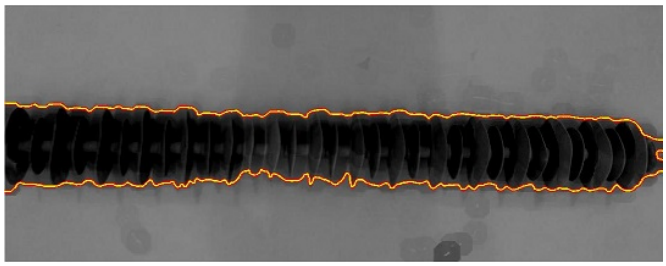


Figure 4.7: The result of the implementation of ACM on Figure 4.5 with curve on insulator boundary. Best viewed in colour.

Figure 4.8: The segmented image of Figure 4.7

4.2.3 Region of Interest

The insulator region of interest is extracted by first obtaining the coordinates and size of the foreground pixels (insulator) from the segmented image, then with same size of segmented image and the original grey scale image, the coordinates and size from the segmented image is mapped into the original grey scale image for placing a minimum bounding box over the entire insulator. Thereafter the insulator ROI is extracted.

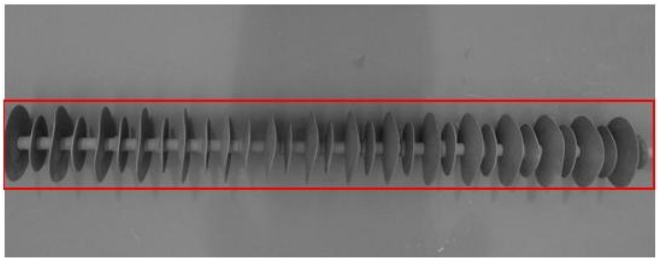


Figure 4.9: Identification of insulator ROI from Figure 4.2. Best viewed in colour.

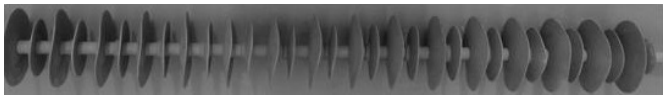


Figure 4.10: Extracted insulator ROI from Figure 4.9

4.2.4 Algorithm of Insulator Segmentation and Region of Interest Extraction

Algorithm 2 gives the steps for insulator segmentation and insulator ROI extraction.

Algorithm 2 Insulator Segmentation and ROI Extraction

Require: I ▷ Source Image
Ensure: E ▷ Extracted Insulator ROI Image
1: Convert the RGB image I into grey scale and Save result as g
2: Define a disk structuring element s
3: Apply Equation 4.1 on g and Save result as T
4: Apply Equation 4.2 on T and Save result as T_d
5: Apply Equation 4.3 and Save result as I_s
6: Initialize a level set function φ
7: **for** n = 1 to maximum value of n **do**
8: Compute $h_1(\varphi)$ and $h_2(\varphi)$ using Equations 3.14 and 3.15 respectively
9: Compute φ^{n+1} by the discretization and linearization of Equation 3.18
10: Check whether curve is stationary
11: **if** curve is stationary **then**
12: Escape from the for loop
13: **end if**
14: **end for**
15: Save segmented image as sIm
16: Extract coordinates of the ROI from sIm
17: Map the coordinates from sIm on g
18: Extract insulator ROI from g and save as E
19: End

4.2.5 Feature Extraction using Local Binary Patterns

LBP presented by Ojala et al. [84] is a texture descriptor for a simple and effective way of texture classification. The LBP operator is defined as [5]:

$$LBP_{(P,R)} = \sum_{P=0}^{P-1} \epsilon(n_p - c_p) * 2^p \tag{4.4}$$

where n_p is the grey values of the circularly symmetric neighbourhood $n_p(p = 0, ..., P - 1)$, c_p is the grey value of the center pixel and $\epsilon(x)$ is defined as:

$$\epsilon(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \tag{4.5}$$

In Equation 4.4, the LBP operator produces different 2^p output values which conform to different 2^p binary patterns produced from the neighbouring P pixels. If the image is rotated, each surrounding pixel in the neighbourhood will move accordingly along the perimeter of the circle, thereby resulting in a different LBP value, except for patterns with $0s$ and $1s$ which remain constant at all rotation angles. To remove the effect of rotation, a rotation-invariant LBP is defined as:

$$LBP^{ri}_{(P,R)} = min\{ROR(LBP_{(P,R),i}|i = 0, 1, ..., P - 1\} \tag{4.6}$$

where ROR is the abbreviation of rotate right. In Equation 4.6, a P-bit number is rotated i times and the minimum value between the resulting numbers for i from 0 to $P1$ is selected. For the modified version of the LBP operator by Ojala et al. [5], a uniformity measure, U, is first of all defined. This is the number of spatial transitions between $1s$ and $0s$ in the pattern. Then patterns that have uniformity measure less than U are defined as uniform patterns. The modified LBP operator for a grey scale and rotation-invariant texture descriptor is defined as:

$$LBP^{riU}_{(P,R)} = \begin{cases} \sum_{P=0}^{P-1} \epsilon(n_p - c_p) * 2^p, & \text{if } U(LBP_{(P,R)}) \\ P + 1, & Otherwise \end{cases} \tag{4.7}$$

where $\epsilon(x)$ is same as in equation 4.5 and $U(LBP_{(P,R)})$ is defined as:

$$U(LBP_{(P,R)}) = |\epsilon(n_{p-1} - c_p) - \epsilon(n_0 - c_p)| + \sum_{P=0}^{P-1} |\epsilon(n_p - c_p) - \epsilon(n_{p-1} - c_p)| \tag{4.8}$$

To compute the LBP histogram using uniform patterns, the LBP histogram accumulates all non-uniform patterns with more than 2 transitions into bin 0, and all other bins of uniform patterns are accumulated in a dedicated bin (see Table 4.1 for simplicity). With the use of $(8, 1)$ neighbourhood, it gives a total of 256 patterns, 58 of which

are uniform, which gives 59 different labels (see Table 4.1) [5]. Then the frequency of occurrence of each number is used to generate a histogram and thereafter normalized. The histogram of all cells are concatenated and gives the feature vector that will be fed into the classifier.

Table 4.1: Computation of uniform LBP labels and the final 59-bin histogram, with "xx" representing the non-uniform patterns

Decimal	Binary	No of Transition	Histogram
xx	xx	> 2	0
0	00000000	0	1
1	00000001	1	2
2	00000010	2	3
.	.	.	.
.	.	.	.
.	.	.	.
254	11111110	1	57
255	11111111	0	58

4.2.6 Classification using Support Vector Machines

The extracted feature vectors are fed into a SVM classifier [114] in order to classify insulators as defectuous and non-defectuous.

Thus, for a labelled set of Z training samples (x_i, y_i) where $x_i \in R^N$ and y_i is the associated label ($y_i \in (-1, +1)$), a SVM classifier finds the optimal hyperplane that correctly separates the training data while maximizing the margin. Then, the training data is required to satisfy the following conditions:

$$\begin{aligned} \varpi \cdot x_i + b \geq +1 \quad y_i = +1 \\ \varpi \cdot x_i + b \leq -1 \quad y_i = -1 \end{aligned} \tag{4.9}$$

This is equivalent to:

$$y_i[\varpi \cdot x_i + b] \geq +1 \quad \forall i = 1, 2, ..., n \tag{4.10}$$

where ϖ is the normal to the hyperplane (weight vector) (Figure 4.11) and b is the bias (scalar). Then the optimal hyperplane is defined as $H : \varpi \cdot x_i + b = 0$ which separates the training data with the maximum margin (Figure 4.11). Also, the points for which the equality in Equation (4.9) holds lie on the plane of which $H_{a1} : \varpi \cdot x_i + b \geq +1$ and $H_{a2} : \varpi \cdot x_i + b \leq -1$ (Figure 4.11). The margin between both hyperplanes (H_{a1} and H_{a2}) equals $2/(||\varpi||^2)$ and $||\varpi||$ is the Euclidean norm of ϖ. The margin $2/(||\varpi||^2)$ is to be maximized or minimized $\frac{1}{2}||\varpi||^2$ subject to the constraints in Equation (4.10).

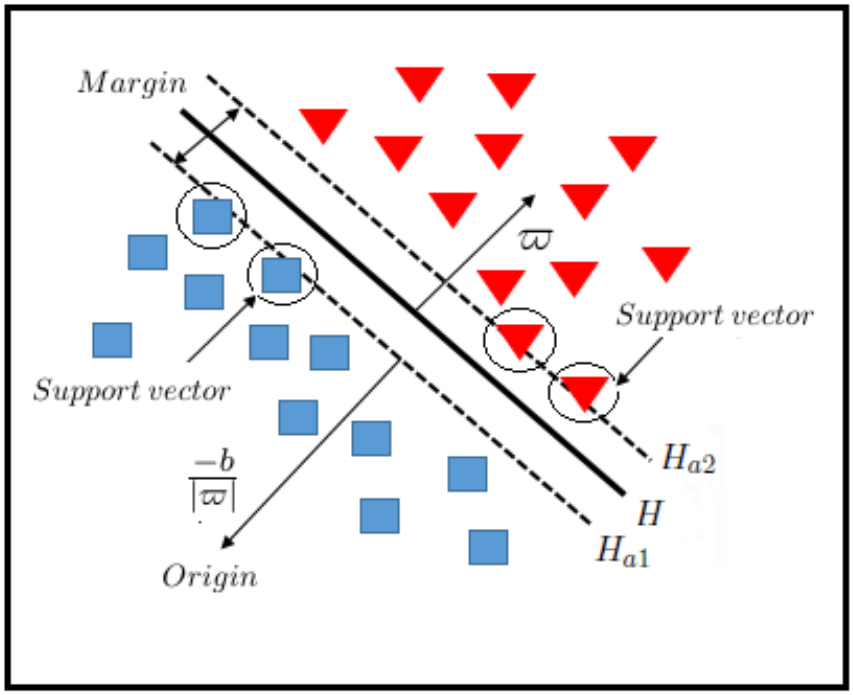


Figure 4.11: Decision boundary or hyperplane and margin of SVM classifier that is linearly separable

Most real life classifications are linearly non separable [148] or contain noise in the dataset [121]. Hence, the application of a soft margin to finding the weight vector using slack variable ζ helps to reduce misclassification. Then, the primal optimization problem is defined as:

$$\begin{aligned} &minimize \\ &\qquad \frac{1}{2}||\varpi||^2 + C\sum_{i=1}^{n}\zeta_i \\ &subject\ \ to \\ &\qquad y_i[\varpi \cdot x_i + b] \geq 1 - \zeta_i, \quad \zeta_i \geq 0\ \forall i \end{aligned} \tag{4.11}$$

where C is a parameter that controls the effect of the margin width and classification error. The primal problem is converted into a dual problem which is a simplified problem [148] and is defined as:

$$\begin{aligned} &maximize \\ &\qquad f(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i\alpha_i\Gamma(x_i \cdot x_j)y_j\alpha_j, \\ &subject\ \ to \\ &\qquad \sum_{j=1}^{n} y_i\alpha_i = 0, \quad and \quad \alpha_i \geq 0, \quad \forall i \end{aligned} \tag{4.12}$$

Equation (4.12) problem is a quadratic function of the α_i subject to linear constraints, which is conveniently solvable by quadratic programming algorithms. The variable α_i is defined such that:

$$\varpi = \sum_{i=1}^{n}\alpha_i y_i x_i \tag{4.13}$$

where the vector ϖ determines the optimal hyperplane and can be written as a linear combination of training vectors. Then, $\Gamma(x_i x_j) \equiv \phi(x_i)^T\phi(x_j)$ is the kernel. Thus, the training vectors x_i are transformed into a higher dimensional space by a function ϕ. There are different kernel functions, these kernels satisfy Mercer's condition which are positive semi definite (it has no non-negative eigenvalues), thus a global optimum is achieved. However, different kernels of SVM were analyzed to determine the best suited for classification. Some of these kernels are: linear kernels of $\Gamma(x_i, x_j) = x_i^T, x_j$, and the Radial Basis Function (RBF) kernel of $\Gamma(x_i, x_j) = exp(-\gamma||x_i - x_j||$, where γ

is the spread of a Gaussian cluster and $\gamma > 0$. With the use of the kernel function, the final SVM classifier becomes:

$$f(\alpha) = sign\left(\sum_{i=1}^{n} y_i \alpha_i \Gamma(x_i, x_j) + b\right) \tag{4.14}$$

4.2.7 Algorithm for the classification of power-line insulator condition

Algorithm 3 gives the steps involved in the classification of power-line insulator condition based on the combination of LBP feature extraction and SVM as a classifier.

Algorithm 3 Classification of power-line insulator condition using LBP and SVM

Require: *Dataset* ▷ Training and test set
Ensure: *Output* ▷ Indicates insulator condition

for each image I in the training set **do**
    Convert each RGB image I into grey scale and save result as g
    Apply LBP operator to g and save result as L
    Compute LBP histogram from L and save result as H
    Use the H to generate the feature vector M
end for
Feed extracted features M into SVM classifier for training
for image T from the test set images **do**
    Apply algorithm 2 to each image T and save result as g
    Partition image g into four parts and save them into I_p, $I_p = \{I_{p1}, I_{p2}, I_{p3}, I_{p4}\}$
    for $n = 1$ to $n = 4$ **do**
        Apply LBP operator on each image save into I_p and save result as $L_p(n)$
        Compute LBP histogram from $L_p(n)$ and save result as $H_p(n)$
        Use the $H_p(n)$ to generate the feature vector $N(n)$
        Compare extracted features of partitioned images $N(n)$ to the trained set M if it matches
        if any label of $N(n)$, has a label that is defectuous **then**
            Output image T is defectuous
        else
            Output image T is non-defectuous
        end if
    end for
end for

4.3 Experimental Results and Discussion

4.3.1 Dataset

To our knowledge, there is presently no publicly available dataset for insulator studies. Hence, the experiment and evaluation are based on the experimental dataset of this research study. The experiment was conducted on a polymeric insulator. The dataset contains 600 insulator images of both defectuous and non-defectuous insulators. The training set is made up of 200 defectuous and 200 non-defectuous insulator images (Figure 4.12). The training set is used to train the extracted features for the model. Also, a validation process is required in addition to the training process. The validation process is implemented in order to avoid over-fitting when the classification parameter needs to be tuned. However, a 10-fold Cross Validation (CV) is applied. The test set is made of 200 insulator images with 100 defectuous and 100 non-defectuous insulator images that are not partitioned. The test set is used to determine the performance characteristics such as the accuracy of the model.

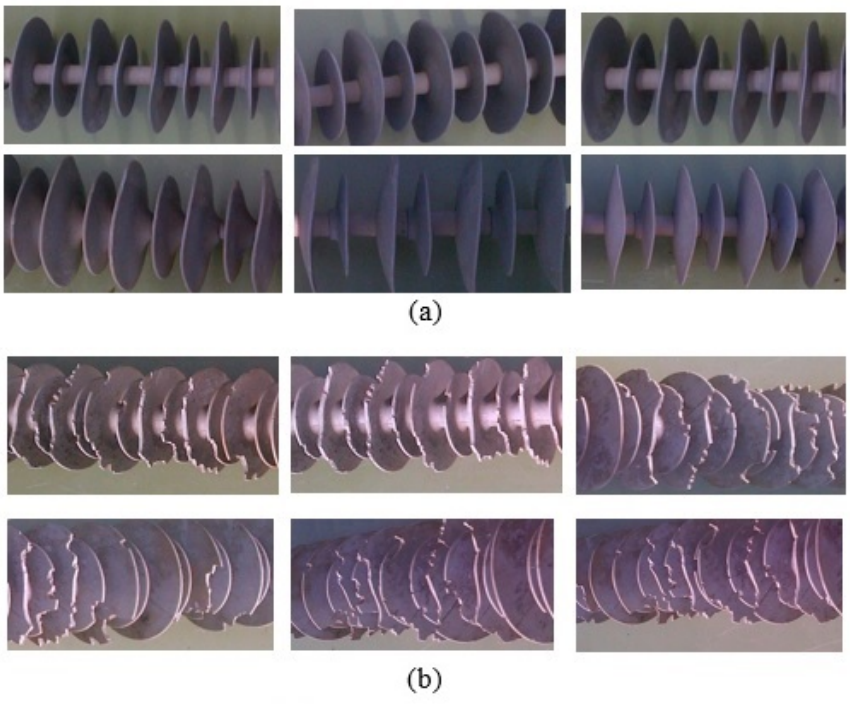


Figure 4.12: Dataset of (a) non-defectious and (b) defectious insulators

4.3.2 Pre-processing

Due to noises (such as shadows and illumination) in the captured images during insulator image acquisition, all insulator images are preprocessed before further analysis. However, in some cases such as in Figure 4.14, it is observed that the insulator brightness is not uniform along the insulator (not a single connected component). Therefore, morphological dilation is applied to Figure 4.14 in order to enhance the brightness of the foreground pixels (insulator) or to have a single connected component as shown in Figure 4.15. Figure 4.16(a) shows the original insulator images and Figure 4.16(b) shows the final processed results after the implementation of morphological top-hat filtering, dilation and image subtraction.

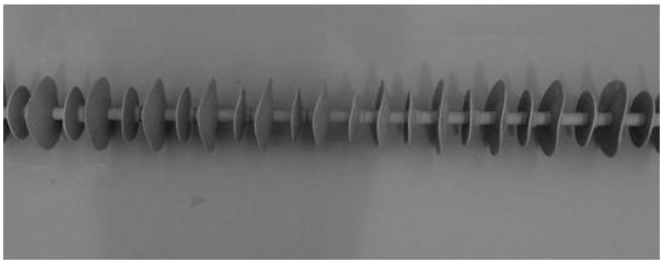


Figure 4.13: Grey-scale image of an insulator

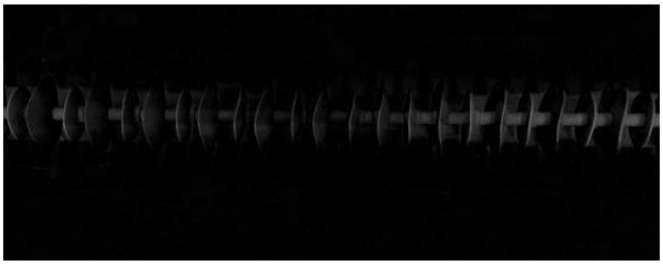


Figure 4.14: Result of morphological top-hat filtering on Figure 4.13

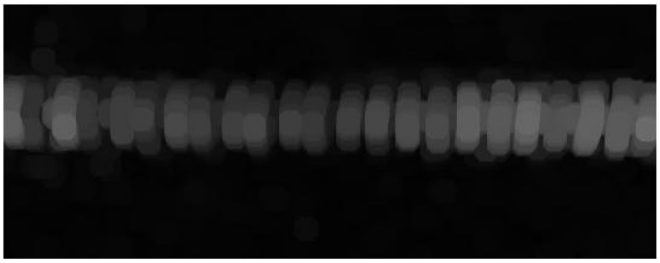


Figure 4.15: Enhanced result on morphological dilation

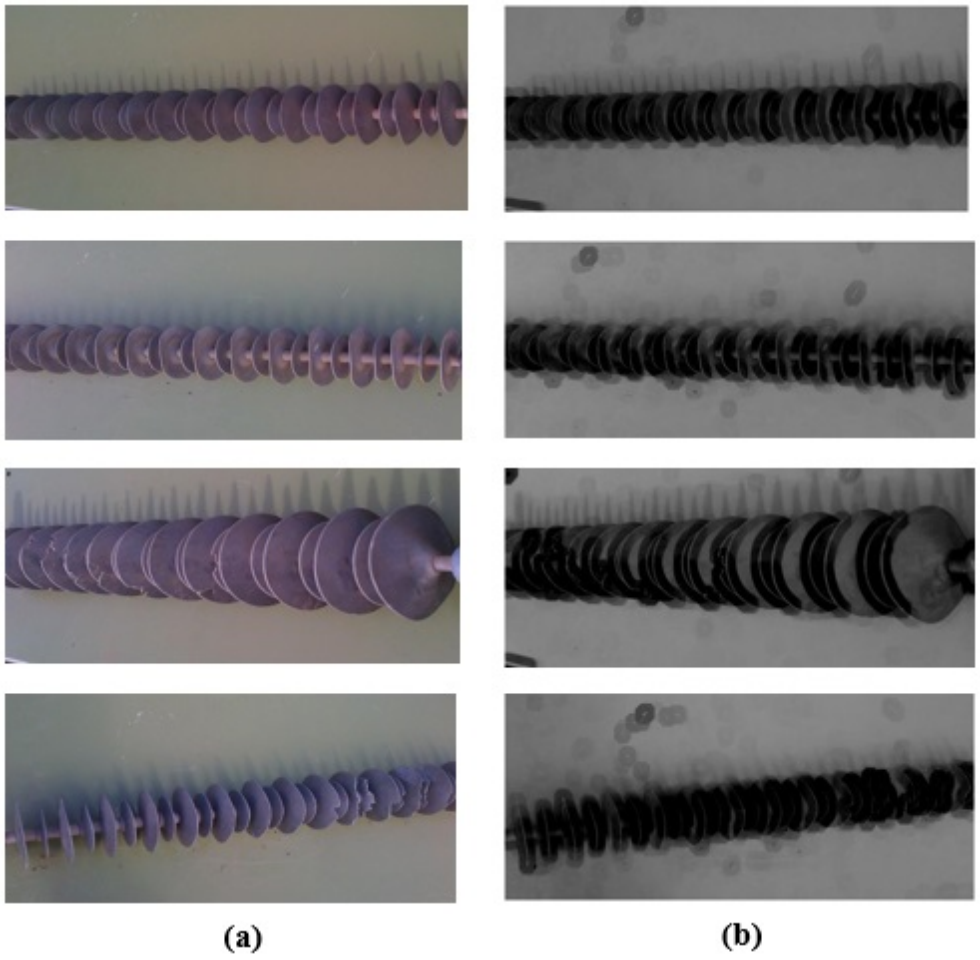


Figure 4.16: (a) Original image and (b) Final processed results; after the implementation of top-hat filtering, dilation and image subtraction on (a).

4.3.3 Segmentation

Some of the final output of the segmented insulator images are shown in Figure 4.17. It is observed that the ACM curve fits on the boundary of the insulators in Figure 4.17(a) and the segmented images are shown in Figure 4.17(b). In Figure 4.17, it is observed that some background pixels are inside the insulator (foreground), as a result post-processing is employed to fill up the holes as shown in Figure 4.18.

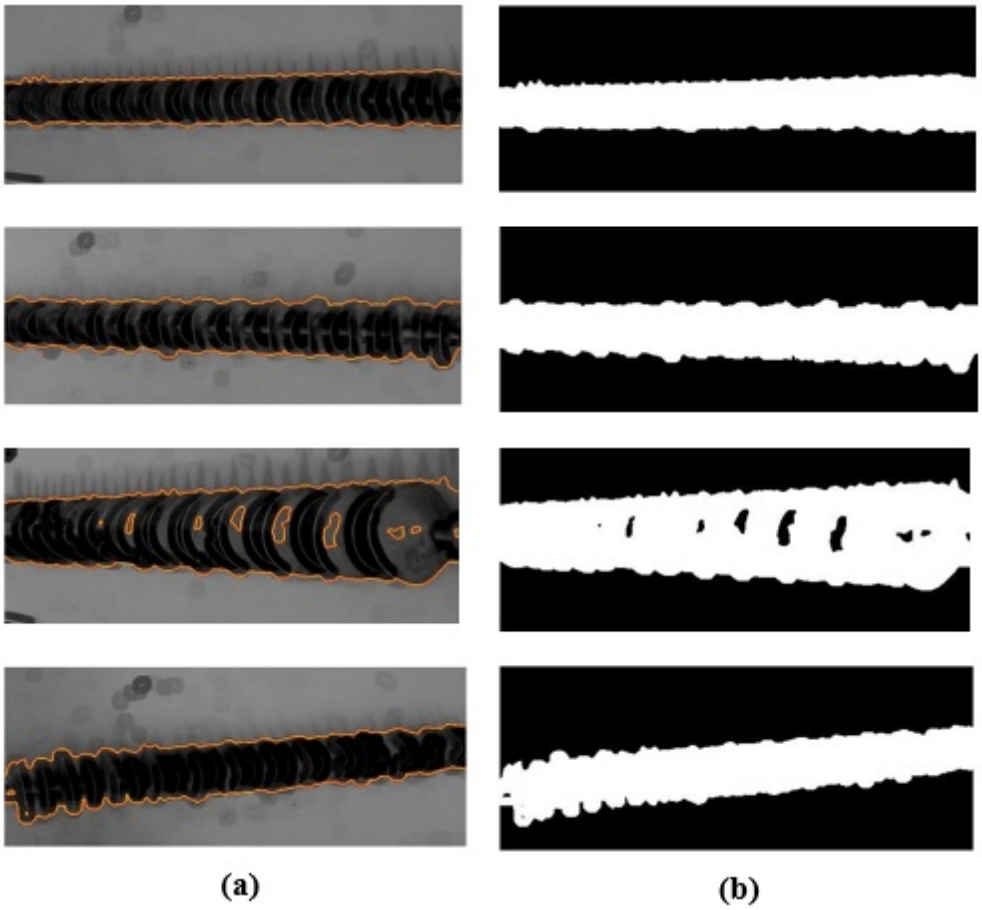


Figure 4.17: (a) Result of the implementation of ACM on Figure 4.16(b) with curve on insulator boundaries and (b) the segmented image of (a). Best viewed in colour.

Figure 4.18: Result of post-processing of (a) Figure 4.17 on row 3 column 2 and (b) row 4 column 2.

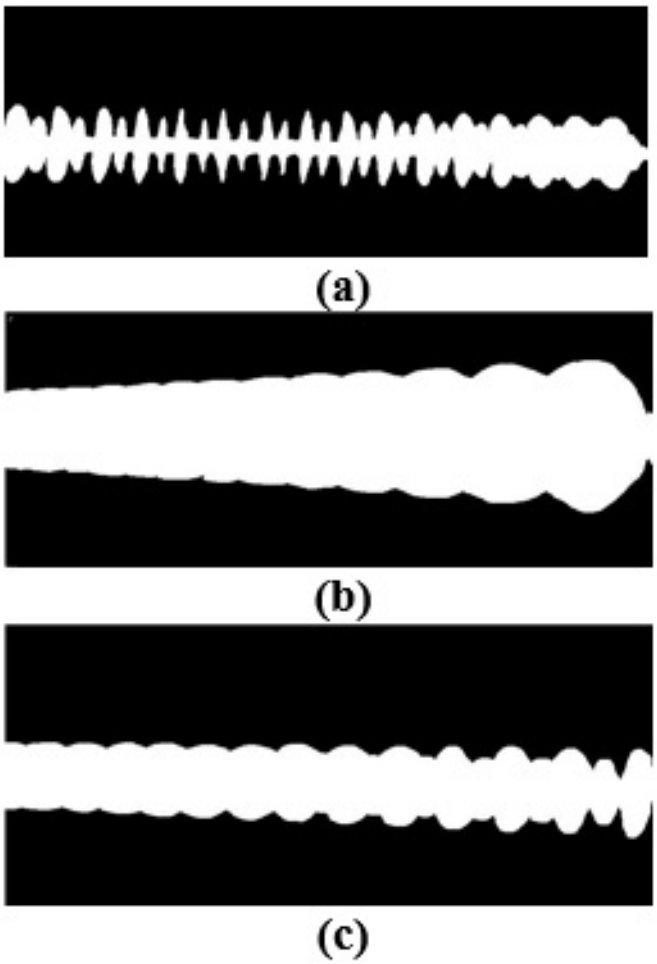


Figure 4.19: Some ground-truth images

In order to evaluate the proposed segmentation method, segmented insulator images created by experts were used (some ground-truth samples are presented in Figure

4.19). Since interest is on the ROI of the insulator, the minimum bounding box over the insulator is required. Therefore, the ground-truth bounding box over the insulator (foreground) is considered as the relevant object and the bounding box of the proposed segmented insulator region is considered as the selected object. Based on this explanation, statistical criteria are employed for performance analysis. The statistical criteria are sensitivity, precision, accuracy and F1-score [149], [150] and are defined as stated in chapter 3; sensitivity also known as recall (Equation 3.23), precision (Equation 3.25), accuracy (Equation 3.26) and F1-score (Equation 3.27). The foreground represents the pixels inside the bounding box or ROI and the background represents the pixels outside the ROI in both segmented images and ground-truth images. All images are in binary for computation. True Positive TP_v is the foreground in the segmented image that overlaps the foreground in the ground-truth image. True Negative TN_v is the background in the segmented image that truly overlaps the background in the ground-truth image. False Positive FP_v is the foreground in the segmented image that are detected as background in the ground-truth image. False negative FN_v is the background in the segmented image that are defined as the foreground in the ground-truth image. Table 4.2 shows the result of the performance analysis.

Table 4.2: Performance analysis

Method	Precision	Recall	F1-Score	Accuracy
Proposed method	87.6%	99.6%	93.0%	94.1%

4.3.4 Insulator Region of Interest Extraction

The extracted coordinate, length and width of the insulator (foreground) image in the segmented image is used to define the insulator ROI in the grey-scale image (Figure 4.20 and 4.21). The red bounding box represents the insulator ROI using the proposed method of segmentation and green bounding box represents the ground-truth. It is observed in Figure 4.20 that both the proposed method of segmentation and the ground-truth overlap each other, while in Figure 4.21, the area of the proposed method is more than the ground-truth. This is due to the under-segmentation caused by the strong edge boundary from the insulator shadow. This reduces the performance of

segmentation. Furthermore, the extracted coordinate, length and width are used to extract the insulator ROI (Figure 4.22) from the original grey-scale image.

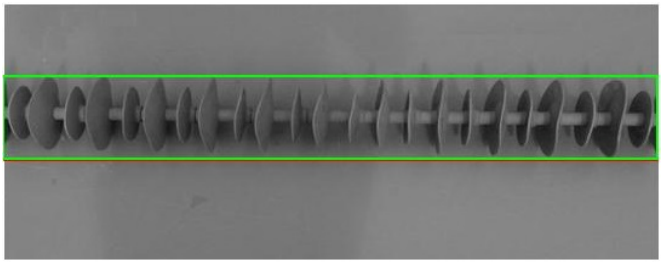


Figure 4.20: Minimum bounding box identified as ROI. The red bounding box represents proposed method of segmentation and green bounding box represents ground-truth. Best viewed in colour

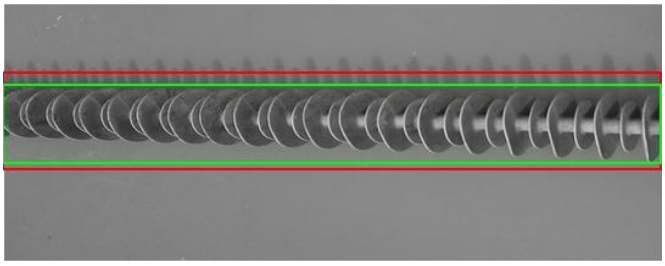


Figure 4.21: Insulator ROI. The red bounding box represents proposed method of segmentation and green bounding box represents ground-truth. Best viewed in colour

Figure 4.22: ROI extracted from Figure 4.13 using the bounding box coordinates obtained from insulator segmentation.

4.3.5 Feature Extraction

Due to the length of each insulator, it will be difficult to capture some smaller regions that are defectuous, therefore each insulator image is partitioned into smaller segments to enhance classification performance. Figure 4.23 shows an insulator image that has been partitioned.

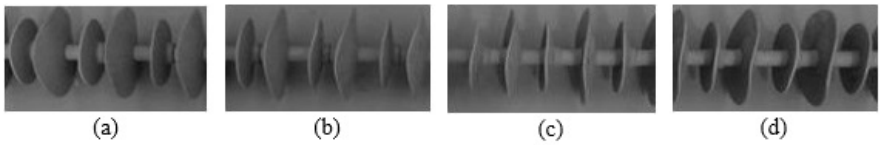


Figure 4.23: Insulator partitioning of Figure 4.22.

The effectiveness of the proposed model is evaluated by comparing it with the GLCM feature extraction implemented in [8]. GLCM is created by computing how often a pixel with the intensity value i, occurs in a specific spatial relationship to a pixel with the intensity value j in an image with number of grey level represented as G_n at a particular displacement distance d and at a specific angle θ. The matrix element $p(i,j|d,\theta)$ contains the second order statistical probability values for changes between intensity values at d and at θ [151]. Let $g(i,j)$ be the $(i,j)^{th}$ entry in the normalized GLCM. The fourteen Haralick features are defined as [152]:

$$Angular\ Second\ Moment\ (ASM) = \sum_{i=0}\sum_{j=0} g(i,j)^2 \tag{4.15}$$

$$Entropy\ (ENT) = -\sum_{i=0}\sum_{j=0} g(i,j) log(g(i,j)) \tag{4.16}$$

$$Contrast\ (CON) = \sum_{n=0}^{G_n-1} n^2 \Bigg\{ \sum_{i=1}^{G_n}\sum_{j=1}^{G_n} g(i,j) \Bigg\}, \quad |i-j| = n \tag{4.17}$$

$$Correlation:\ (COR) = \frac{\sum_{i=0}^{G_n-1}\sum_{j=0}^{G_n-1} p(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y} \tag{4.18}$$

where μ_x, μ_y, σ_x and σ_y are the means and standard deviations of the partial probability functions, g_x and g_y.

$$Sum\ of\ Squares:\ Variance\ (VAR) = \sum_{i=0}\sum_{j=0} (i-\mu)^2 g(i,j) \tag{4.19}$$

$$Sum\ Average:\ (SAV) = \sum_{i=2}^{2G_n-2} (i)(g_{x+y}(i)) \tag{4.20}$$

where x and y are the coordinates along the row and column of the co-occurence matrix and $g_{x+y}(i)$ is the probability of co-occurence matrix coordinates adding to $x + y$.

$$Sum\ Entropy:\ (SEN) = -\sum_{i=2}^{2G_n-2} g_{x+y}(i) log\{g_{x+y}(i)\} \tag{4.21}$$

$$Sum\ Variance:\ (SVA) = \sum_{i=2}^{2G_n-2} (1-f)^2 (g_{x+y}(i)) \tag{4.22}$$

where f is equal to the equation of sum entropy (Equation 4.21.

$$Inverse\ Difference\ Moment:\ (IDM) = \sum_{i=0}\sum_{j=0} \frac{1}{1+(i-j)^2} g(i,j) \tag{4.23}$$

$$Difference\ Variance:\ (DVA) = \sum_{i=0}^{G_n-1} i^2 g_{x-y}(i) \tag{4.24}$$

$$Difference\ Entropy:\ (DEN) = -\sum_{i=0}^{G_n-1} g_{x-y}(i) log\{g_{x-y}\} \tag{4.25}$$

$$Information\ Measure\ of\ Correlation\ 1:\ (IMC1) = \frac{AXY - AXY1}{max\{AX, AY\}} \tag{4.26}$$

where, AX and AY are entropies of g_x and g_y,

$$AXY = \sum_{i=0} - \sum_{j=0} g(i,j) log(g(i,j))$$

$$AXY1 = \sum_{i=0} - \sum_{j=0} g(i,j) log\{g_x(i) g_y(j)\}$$

$$Information\ Measure\ of\ Correlation\ 2:\ (IMC2) = (1 - exp[-2(AXY2 - AXY)])^{1/2} \tag{4.27}$$

where,

$$AXY1 = \sum_{i=0} - \sum_{j=0} g_x(i)g_y(j)log\{g_x(i)g_y(j)\}$$

$$Maximum\ Correlation\ Coefficient:\ (MCC) = Square\ root\ of\ the\ largest\ eigenvalue\ of\ Q \tag{4.28}$$

where,

$$Q(i,j) = \sum_{k} \frac{g(i,k)g(j,k)}{g_x(i)g_y(k)}$$

4.3.6 Classification

Tables 4.3 and 4.4 show the condition of insulators based on 10-fold cross validation for the proposed model and GLCM [8] respectively. Tables 4.5 and 4.6 show the performance of insulator condition using the proposed model and GLCM [8] respectively based on the images in the test set.

The performance analysis of the proposed method is measured using Equation 3.26. The accuracy of the proposed method and GLCM [8] method using the testing set images are 85.5% and 80.0% respectively (Table 4.7). Hence, a better performance is obtained when LBP features are extracted and fed into SVM classifier over GLCM features fed into SVM.

Table 4.3: Confusion matrix for the CV using LBP with SVM

	Defectuous	Non-defectuous
Defectuous	200	0
Non-defectuous	0	200

Table 4.4: Confusion matrix for the CV using GLCM [8] with SVM

	Defectuous	Non-defectuous
Defectuous	188	12
Non-defectuous	9	191

Table 4.5: Confusion matrix of the test set using LBP with SVM

	Defectuous	Non-defectuous
Defectuous	94	6
Non-defectuous	23	77

Table 4.6: Confusion matrix of the test set using GLCM [8] with SVM

	Defectuous	Non-defectuous
Defectuous	87	13
Non-defectuous	27	73

Table 4.7: Accuracy of GLCM [8] and LBP with same SVM classifier (%)

	10-fold validation ACC (%)	Testing ACC (%)
GLCM [8]	94.8%	80.0%
Proposed method	100%	85.5%

It can be observed from Tables 4.3 and 4.4, that LBP fed into SVM outperformed GLCM features fed into SVM in the application of 10-fold cross validation using the training set. With the application of test images, the accuracy of both methods dropped. The LBP has a better performance over GLCM. This is because the LBP is rotation invariant unlike the GLCM. Moreover it is just one direction parameter during computation.

In the classification phase, SVM with radial basis function kernel was used. A Cross Validation (CV) process was performed in order to predict or optimize the model parameter (such as the regularization parameter and kernel width) to fit the training set. A k-fold cross-validation randomly partitions the training set into k-equal sized subsets, whereby a single subset is retained as a test set and all other subsets are used as the training set. Then the cross-validation process is repeated, based on the number of fold (k) times, with the k subset used once as the validation set and thereafter averaging

the k results of all folds to obtain a single result. However, the number of fold is varied from 2 to 10 and the accuracy is computed using Equation 3.26. The result is shown in Figure 4.24.

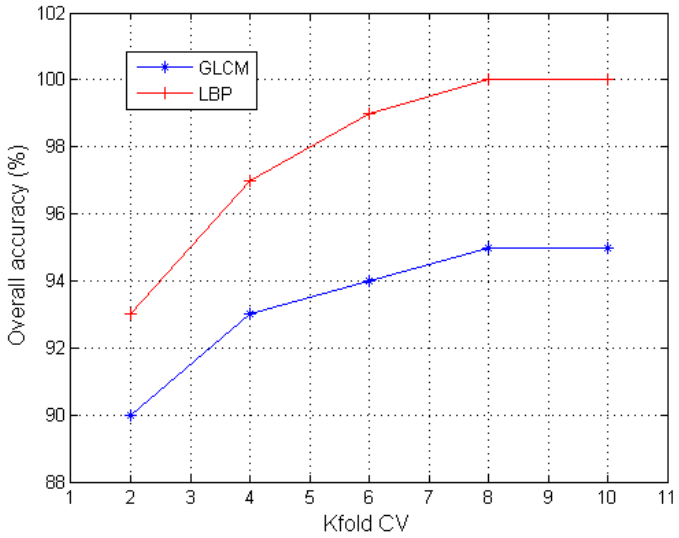


Figure 4.24: Accuracy on cross-validation using SVM

In Figure 4.24, the accuracy of both methods increases from k = 2 to k = 10. However, if 5% is considered as the threshold of the error rate (1-accuracy(%)), then it can be stated that a better fitting model to the training set is achieved from k = 8 to 10 for both methods. There is an increase in both methods because of the principle of k-fold CV. For example, at k = 2, it partitions the training set into 2 equal sizes, 50:50, which means that it trains with one part of the 50 and test with the other part of 50. For k = 10, it partitions the training set into 10 equal sizes, 20:80, and used the 80 for training while the 20 for testing. Comparing both cases of k = 2 and k = 10, it means that more images will be used for training for k = 10 than k = 2, as a result having a better chance to be able to identify unknown image(s).

Figure 4.25 shows the overall accuracy when the training set is varied from 100, 200, 300, 400, along side the test set from 500, 400, 300, 200 respectively for an SVM classifier using LBP and GLCM feature extraction methods. Each pair (training set and test set) amount to 600 images in all cases of varying the size of the dataset. The

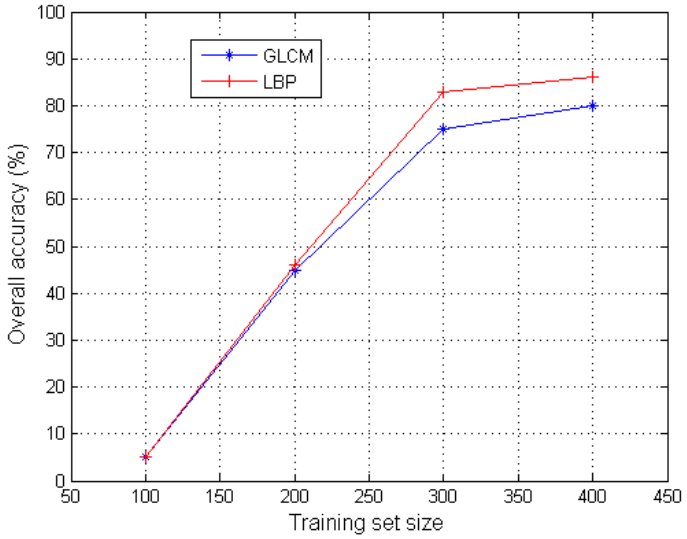


Figure 4.25: Accuracy on test set using SVM

increasing training set size increases the accuracy and decreases the error rate. It can be observed in Figure 4.25, that there is a sharp rise in the training set when increased from 100 to 300. This is because the training set is not large enough. When the training set is increased above 300, a gradual increase is noticed. At this point forward, there is also a reduced error rate which may be approximately the same within this region and the model performs best at this region.

4.4 Statistical Test

In order to validate the accuracy of the methods, the McNemar's test introduced by Quinn McNemar in 1947 is used. The McNemar's test is a statistical test used on paired (matched) data on a dichotomous item [153], [154]. Let A be the combination of GLCM with SVM algorithm and B be the combination of LBP with SVM algorithm. Given the following number of cases, Table 4.8 shows that:

- Number of insulators estimated defectuous for A and defectuous for B = a
- Number of insulators estimated non-defectuous for A and defectuous for B = b
- Number of insulators estimated defectuous for A and non-defectuous for B = c

- Number of insulators estimated non-defectuous for A and non-defectuous for B = d

Table 4.8: Contingency table

		A	
		Defectuous	Non-defectuous
B	Defectuous	a	b
	Non-defectuous	c	d

A contingency table is generated from Tables 4.5 and 4.6 based on [154] and it is shown in Table 4.9.

Table 4.9: Contingency table

		A		
		Defectuous	Non-defectuous	
B	Defectuous	181	19	$t_1 = 200$
	Non-defectuous	50	150	$t_2 = 200$
		$t_3 = 231$	$t_4 = 169$	$T = 400$

The McNemar's test uses data from the two discordant entries b and c [153] from Table 4.8, where n is the total number of matched pairs ($n = a + b + c + d$). The McNemar's test is used to determine whether the null hypothesis H_o is accepted or rejected. The H_o states that there is no significant difference between the two algorithms and alternative hypothesis H_a states that there is a significant difference between both algorithms. The McNemar's test is computed using Equation 4.22 [154].

$$\chi^2 = \frac{(|b - c| - 1)^2}{b + c} \tag{4.29}$$

where χ^2 is chi-squared distribution with one degree of freedom (DoF).

Therefore, the computed chi-square value (χ^2) using Equation 4.29 is 13.04. By convention, scientists often use a P value of 0.05 (5%) for deviation of significant test. However, if it is greater, then the null hypothesis H_o is rejected. The P value of 0.05 for 1 DoF in the chi-square (χ^2) distribution table is 3.841 [155]. Since the computed

$\chi^2 > 3.841$, H_o is rejected. Therefore, it means that the accuracy of the two algorithms (A and B) are statistically different.

4.5 Conclusion

The classification of power-line insulator condition has been investigated and presented. The segmentation method was evaluated using manually created ground-truth by experts. The comparison of LBP and GLCM feature extraction methods has also been presented. It is shown that LBP outperformed GLCM when both features are fed into a SVM classifier while considering their accuracy rates. From the experiment, it is evident that the LBP is statistically robust, more stable and less prone to noise. It is also observed that variation in the training set size resulted in increased accuracy with a reduced error rate. It is observed from the McNemar's test, that there is significant difference between LBP and GLCM using the same SVM classifier.

Chapter 5

Classification of Insulator Condition using Scale Invariant Feature Transform with K-Nearest Neighbour

5.1 Introduction

This chapter introduces an automated method for the classification of power-line insulator condition using SIFT with KNN. Before the application of the SIFT algorithm for the extraction of features, the insulator region of interest is extracted from the image. The extraction of the insulator is done using the active contour model, followed by the extraction of the insulator ROI. The SIFT algorithm is used to extract features that are stable and invariant by translation, rotation, scale and stable to changes in the illumination. This is then fed into a KNN classifier to determine the condition of insulators. The SIFT algorithm discussed in this chapter is organized into four major phases namely, scale space extrema detection, keypoint localization, orientation assignment and keypoint descriptor generation. The developed automated model based on computer vision will be able to identify and analyze insulators condition safely, speedily and accurately-better than the manual or traditional method. Also, the proposed method will be able to minimize the failure caused by insulator defect which has a major impact on the transmission and distribution of electricity.

The flowchart of the proposed method of classification of insulator condition using SIFT with KNN is shown in Figure 5.1.

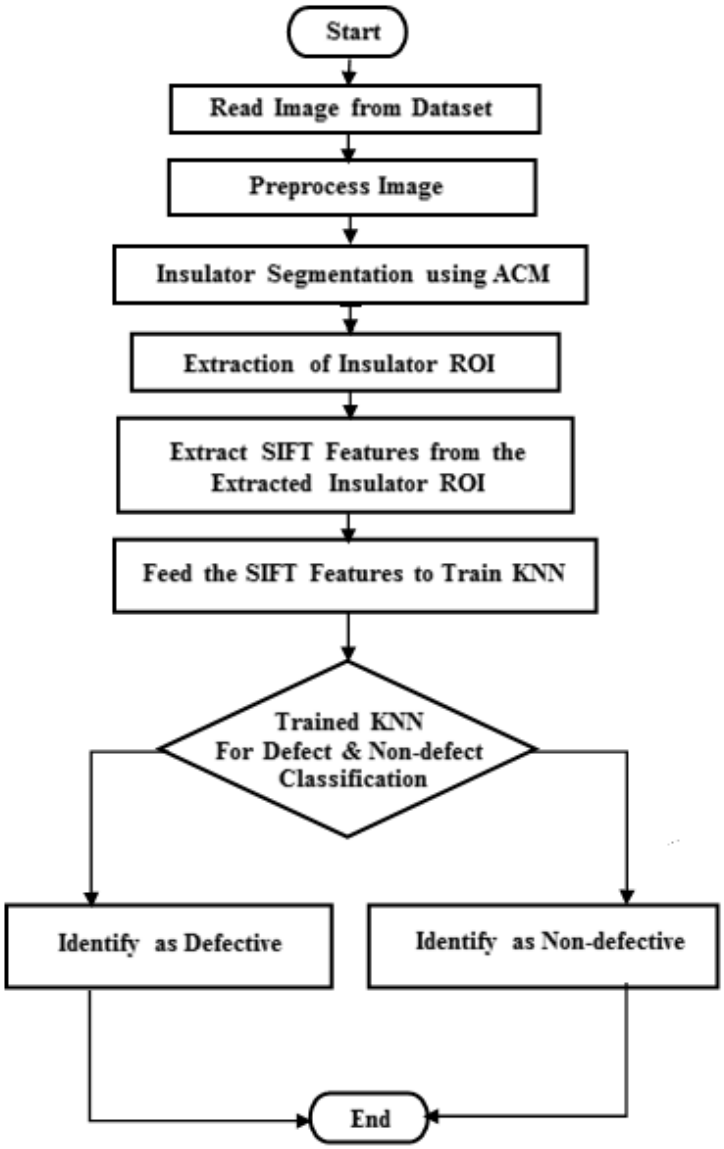


Figure 5.1: Flowchart of SIFT and KNN

5.2 Proposed Method

5.2.1 Insulator Segmentation and Region of Interest Extraction

This section is based on the same principle employed in pre-processing, insulator segmentation and region of interest extraction presented in Chapter 4. In order to reduce noise and improve the quality of images, pre-processing and post-processing steps are required. RGB images are converted into grey scale images. Then, morphological operation is applied to the grey scale images for the purpose of image enhancement (see Section 4.2.1 in Chapter 4). Insulator images are enhanced using the top-hat filter. Then, the result obtained from the application of top-hat filtering is further enhanced using the morphological operation known as dilation. The output image of the morphological dilated image is then subtracted from the original grey level insulator image for further analysis on insulator segmentation (see Section 4.2.1 in Chapter 4 for detailed illustration).

The region-based ACWE by Chan-Vese is used for insulator segmentation and it is based on Equation 3.13 in Chapter 3 with the same parameter settings and principle of operation.

Insulator region of interest is extracted by first obtaining the coordinates and size of the foreground pixels (insulator) from the segmented image. With the same size of segmented image and the original grey scale image, the coordinates and size from the segmented image are mapped into the original grey scale image for placing a minimum bounding box over the entire insulator; thereafter is the extraction of the insulator ROI.

5.2.2 Algorithm of Insulator Segmentation

Algorithm 4 gives the steps for insulator segmentation and ROI extraction.

Algorithm 4 Insulator Segmentation and ROI Extraction

Require: I ▷ Source Image
Ensure: E ▷ Extracted Insulator ROI Image

Convert the RGB image I into grey scale and Save result as g
Define a disk structuring element s
Apply Equation 4.1 on g and Save result as T
Apply Equation 4.2 on T and Save result as T_d
Apply Equation 4.3 and Save result as I_s
Initialize a level set function φ
for n = 1 to maximum value of n **do**
    Compute $h_1(\varphi)$ and $h_2(\varphi)$ using Equations 3.14 and 3.15 respectively
    Compute φ^{n+1} by the discretization and linearization of Equation 3.18
    Check whether curve is stationary
    if curve is stationary **then**
        Escape from the for loop
    end if
end for
Save segmented image as sIm
Extract coordinates of the ROI from sIm
Map the coordinates from sIm on g
Extract insulator ROI from g and save as E
End

5.2.3 Feature Extraction using Scale Invariant Feature Transform

This section discusses the extraction of the local keypoints descriptor using the SIFT algorithm. The features are less sensitive to scale, translation, and rotation. Figure 5.2 describes the data flow of the SIFT algorithm traversing from one stage to another. The stages of the SIFT algorithm are outlined and described in the following subsections.

5.2.3.1 Scale Space Extrema Detection

The proper identification of a real world object depends on the scale representation of the object. The scale space is the first stage of the SIFT algorithm which searches all scales over several octaves of image. It is implemented by using Difference of Gaussian (DoG) to identify potential keypoints that are invariant to scale and image rotation. In the scale space extrema detection phase, the SIFT algorithm computes the “scale”, “DoG”, and “extrema” over several “octaves”. The computation of scale space is defined as:

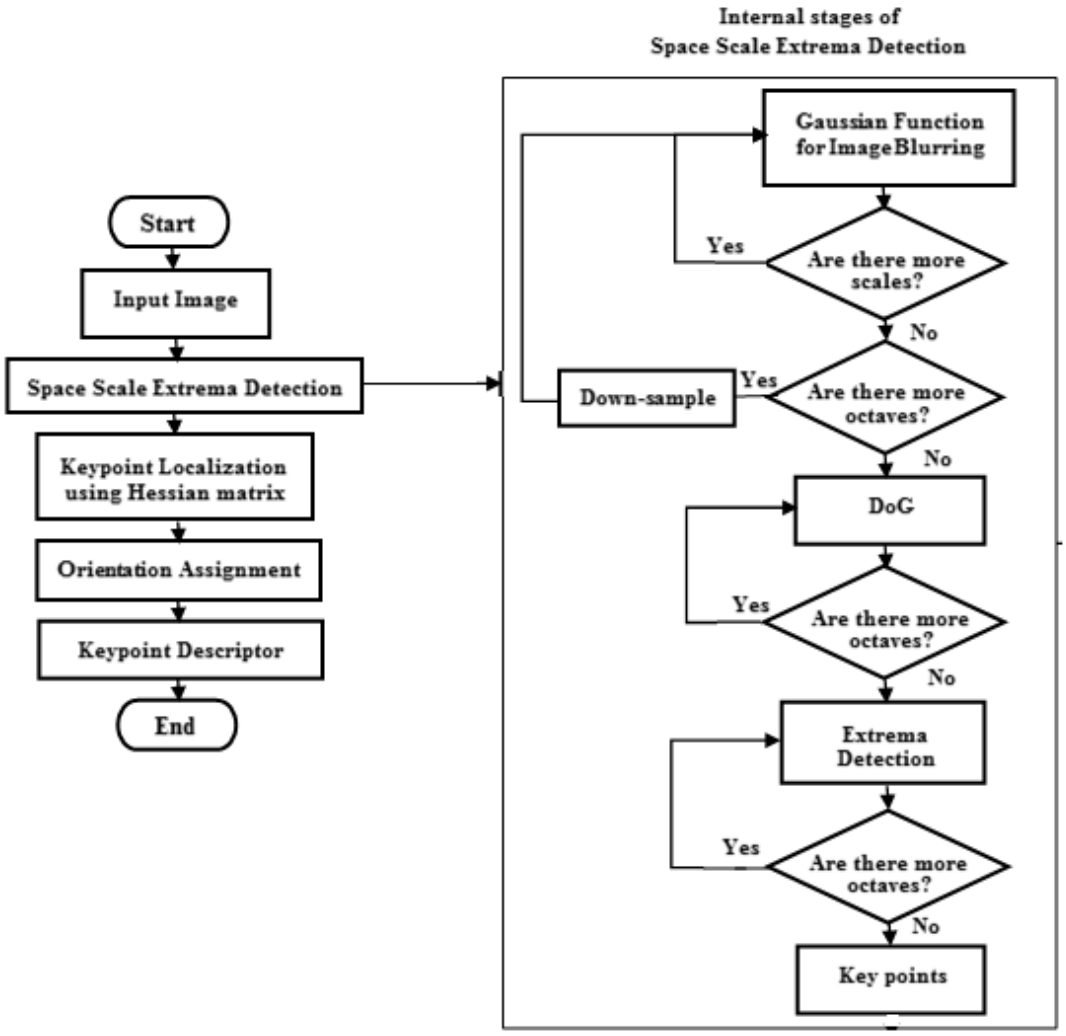


Figure 5.2: SIFT algorithm for data flow

$$S(x, y, \sigma) = G_\alpha(x, y, \sigma) * I(x, y) \tag{5.1}$$

where $I(x, y)$ is the image, $*$ denotes convolution operator for two-dimensional at point x, y and $G_\alpha(x, y, \sigma)$ represents the Gaussian function for image blurring and it is defined as $G_\alpha(x, y, \sigma) = Ke^{-\rho}$, where $K = 1/(2\pi\sigma^2)$ and $\rho = (x^2 + y^2)/(2\sigma^2)$.

In general, the k^{th} scale of the image, for $k \geq 1$ is defined using Equation 5.1. Therefore, computing the scale for each image point $I(x, y)$, a scalar product is applied

between the pixel $I(x, y)$ and a Gaussian weighted window gxg is placed over that point. The SIFT algorithm repeatedly computes the scale of the image and then produces a sequence of scales known as octave [156]. In the next step, all these octaves are used to generate the DoG images. The DoG can be computed by subtracting each adjacent image with a multiplication factor of m and it is derived as:

$$\begin{aligned} D(x, y, \sigma) &= (G_\alpha(x, y, m\sigma) - G_\alpha(x, y, \sigma)) * I(x, y) \\ &= S(x, y, m\sigma) - S(x, y, \sigma) \end{aligned} \tag{5.2}$$

For example, Let $S_0^0 = I$ represents the original image, so that S_j^i represents sequences of scales (octave), the superscript j represents the level of octave (where "n" octave, $0 < j \leq n - 1$) and subscript i represents the sequence of scales (where $0 \leq i < k + 3$, and $k \geq 1$ is an integer). Therefore, for an image $I(x, y)$, $S_0^{i+i}(x, y) = G_\alpha(x, y, \sigma_i) * S_0^i(x, y)$, where $0 \leq i < k + 3$, the algorithm computes the scale as:

$$S_0^0.G_\alpha \rightarrow \ S_1^0.G_{\alpha_1} \rightarrow \ S_2^0 \cdots \ S_k^0.G_{\alpha_k} \rightarrow \ S_{k+1}^0.G_{\alpha_{k+1}} \rightarrow \ S_{k+2}^0.G_{\alpha_{k+2}} \rightarrow \ \ S_{k+3}^0 \tag{5.3}$$

The sequence of scales in Equation 5.3 is referred to as an octave. For the next octave, a reduction in image resolution is required. The resolution of an image can be reduced by a factor of 2 in each dimension by sampling every other pixel of the image. Then the next octave (second octave) similar to the first octave (octave zero) (Equation 5.3) can be defined as:

$$S_0^1.G_\alpha \rightarrow \ S_1^1.G_{\alpha_1} \rightarrow \ S_2^1 \cdots \ S_k^1.G_{\alpha_k} \rightarrow \ S_{k+1}^1.G_{\alpha_{k+1}} \rightarrow \ S_{k+2}^1.G_{\alpha_{k+2}} \rightarrow \ \ S_{k+3}^1 \tag{5.4}$$

For "n" octaves with a reduction in each dimension by a factor of 2, it is defined as:

$$S_0^j.G_\alpha \rightarrow \ S_1^j.G_{\alpha_1} \rightarrow \ S_2^j \cdots \ S_k^j.G_{\alpha_k} \rightarrow \ S_{k+1}^j.G_{\alpha_{k+1}} \rightarrow \ S_{k+2}^j.G_{\alpha_{k+2}} \rightarrow \ \ S_{k+3}^j \tag{5.5}$$

Then the DoG can be computed over the fixed octave "j" and for $0 \leq i < k + 3$ as:

$$S_1^j - S_0^j, \ \ S_2^j - S_1^j, \ \cdots \ S_{k+2}^j - S_{k+1}^j, \ \ S_{k+3}^j - S_{k+2}^j \tag{5.6}$$

where the difference is for each pair of corresponding pixels S_{i+1}^{j} and S_{i}^{j}. This can be written for an i^{th} DoG as:

$$D_{i}^{j} = S_{i+1}^{j} - S_{i}^{j} \tag{5.7}$$

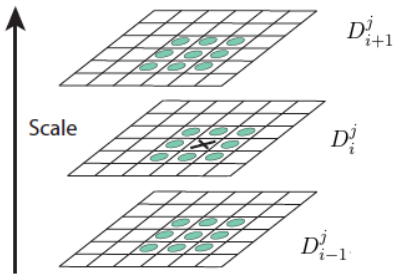


Figure 5.3: Extrema detection [6].

For an extrema detection, lets consider three sets of adjacent DoG images as D_{i-1}^{j}, D_{i}^{j} and D_{i+1}^{j} in an octave j (Figure 5.3). Let element $D_{i}^{j}(x, y)$ represents the "X" mark in Figure 5.3 with 26 neighbouring pixel shown in green circle in Figure 5.3. Thus, element $D_{i}^{j}(x, y)$ is regarded as an extremum if the pixel value is larger than all the neighbouring pixels or if it is smaller than it neighbouring pixels.

5.2.3.2 Keypoint Localization

After the comparison of a pixel with its neighbouring pixels to obtain keypoint candidate (extrema), the next step is to perform an interpolation of nearby data for location, scale, and ratio of principal curvatures. This information allows points to be rejected which have low contrast or that are poorly localized along an edge. The interpolation is performed using the quadratic Taylor expansion of the DoG scale-space function, $D_(x, y, \sigma)$ with the keypoint candidate as the origin. This Taylor expansion is defined as:

$$D(\chi) = D + \frac{\partial D^{T}}{\partial \chi}\chi + \frac{1}{2}\chi^{T}\frac{\partial^{2} D}{\partial \chi^{2}}\chi \tag{5.8}$$

where D and its derivatives are evaluated at the keypoint candidate and $\chi = (x, y, \sigma)^T$ is the offset from this point. The location of the extremum, $\tilde{\chi}$ is determined by taking the derivative of this function with respect to χ and setting it to zero, giving $\tilde{\chi} = -\frac{\partial^2 D^{-1}}{\partial \chi^2} \frac{\partial D}{\partial \chi}$. Then, if the offset $\tilde{\chi}$ is larger than a threshold in any dimension, then it indicates that the extremum lies closer to a different keypoint candidate. Therefore the keypoint candidate is changed and the interpolation is carried out about that point, otherwise the offset $\tilde{\chi}$ is added to its keypoint candidate to get the interpolated estimate for the location of the extremum.

To reject low contrast points, the function value at the extremum $D(\tilde{\chi})$, is used to discard all extrema with a value of $|D(\tilde{\chi})|$ less than a threshold value, and it is defined as:

$$D(\tilde{\chi}) = D + \frac{1}{2} \frac{\partial D^T}{\partial \chi} \chi \tag{5.9}$$

In order to increase stability, since rejection of keypoints with low contrast is not a sufficient means, keypoints are eliminated with poorly determined locations but with high edge responses. Thus, to detect the extrema on edges, the principal curvature is used which is computed from a 2x2 Hessian matrix, H, at the location and scale of the keypoint. The 2x2 Hessian matrix is defined as:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{5.10}$$

Then, the trace and determinant of H are computed for the generation of the ratio of principal curvature. This quantity is compared with a threshold value to determine if an extremum is to be discarded or not. After the elimination of extrema points, the remaining points are called keypoints.

5.2.3.3 Orientation Assignment

At this stage, the main aim is to achieve invariance to image rotation. The magnitude $m(x, y)$ and rotation $\theta(x, y)$ for each pixel in a neighbouring region around the keypoint in the Gaussian smoothed image $S(x, y)$ at this scale, can be computed as:

$$m(x,y) = \sqrt{(S(x+1,y) - S(x-1,y))^2 + (S(x,y+1) - S(x,y-1))^2} \tag{5.11}$$

$$\theta(x,y) = tan^{-1}((S(x,y+1) - S(x,y-1))/(S(x+1,y) - S(x-1,y))) \tag{5.12}$$

Then an orientation histogram is generated that has 36 bins covering a 360 degree range of orientations. The peaks in the histogram correspond to the pre-eminent orientations of local gradient and the orientation is detected as the highest peak in the histogram. For any other local peak that is within 80% of the highest peak, a keypoint is created with that orientation.

5.2.3.4 Keypoint Descriptor Generation

In this stage, a descriptor vector is computed for each keypoint such that the descriptor is highly distinctive and partially invariant to illumination and 3D viewpoint. The descriptor vector is computed as a set of orientation histograms on 4x4 pixel neighbourhoods with 8-bins. These histograms are computed from the magnitude and orientation values of a 16x16 region around the keypoint, with each histogram having a 4x4 subregion of the original region. The magnitudes are further weighted using a Gaussian function and the width of the descriptor window is based on the value of sigma in the Gaussian function. With a 4x4 = 16 histogram, with each histogram having 8-bins, the vector has 128 elements. Thus, for invariance to affine changes in illumination, the vector is normalized to a unit length.

5.2.4 Classification using K-Nearest Neighbour

The extracted feature vectors are fed into a KNN classifier in order to classify insulators as defectuous and non-defectuous. Typically, KNN is used to train a set of feature vectors or attribute vectors with a given corresponding class label in order to predict a class of an unknown instance x. An instance x relates to a point in a n-dimensional space and can be represented by an attribute vector $[v_1(x), v_2(x), ..., v_n(x)]$, where n is the number of attributes. In our case, KNN uses the Euclidean distance to measure the distance between instance x_i and x_j. The Euclidean distance is defined in [111], [112], [113] as:

$$d(x_i, x_j) = \sqrt{\sum_{m=1}^{n} (v_m(x_i) - v_m(x_j))^2} \tag{5.13}$$

Given a new instance y, KNN uses the k-nearest instance in the training set i.e. $x_1, x_2, ..., x_k$, thereby returning the result of classifying y as defined in equation 5.14.

$$c(y) \leftarrow \arg\max_{c \in C} \sum_{r=1}^{k} \delta(c, c(x_i)) \tag{5.14}$$

where $c(y)$ is the class of the instance y, k is the number of neighbours, C and c represents the class variable and $\delta(c, c(x_i))$ is equal to 1, if c is equal to $c(x_i)$ and 0 otherwise.

5.2.5 Algorithm for the classification of power-line insulator condition

Algorithm 5 gives the steps involved in the classification of power-line insulator condition using SIFT with KNN.

Algorithm 5 Classification of power-line insulator condition using SIFT and KNN

Require: *Dataset* ▷ Training and test set
Ensure: *Output* ▷ Indicates insulator condition

for each image I in the training set **do**
    Convert each RGB image I into grey scale and save result as g
    Generate a scale space from g using Equation 5.1
    Find key-points or interest points using Equation 5.2
    Reject low contrast and bad key-points using Equations 5.9 and 5.10
    Assign orientation to the key-points using Equations 5.11 and 5.12
    Generate SIFT key-point descriptor as D
end for
Feed extracted features D into KNN classifier for training
for image T from the test set images **do**
    Apply algorithm 4 to each image T and save result as g
    Partition image g into four parts and save them into I_p, $I_p = \{I_{p1}, I_{p2}, I_{p3}, I_{p4}\}$
    for $n = 1$ to $n = 4$ **do**
        Generate a scale space from each image saved into I_p using Equation 5.1
        Find key-points or interest points using Equation 5.2
        Reject low contrast and bad key-points using Equations 5.9 and 5.10
        Assign orientation to the key-points using Equations 5.11 and 5.12
        Generate SIFT key-point descriptor as D
        Save each D from I_p result as $S_p(n)$
        Use the $S_p(n)$ to generate the feature vector $N(n)$
        Compare extracted features of partitioned images $N(n)$ to the trained set M if it matches
        if any label of $N(n)$, has a label that is defectuous **then**
            Output image T is defectuous
        else
            Output image T is non-defectuous
        end if
    end for
end for

5.3 Experimental Results and Discussion

5.3.1 Dataset

To our knowledge, there is presently no publicly available dataset for insulator studies and therefore the dataset used in this chapter is the same as presented in Chapter 4.

5.3.2 Pre-processing

Due to a high signal to noise ratio such as shadows and illumination in the captured images, all insulator images were pre-processed before further analysis. Figure 5.4(a)

shows the original grey-scale images; Figures 5.4(b) - (d) show the results of the pre-processed stages.

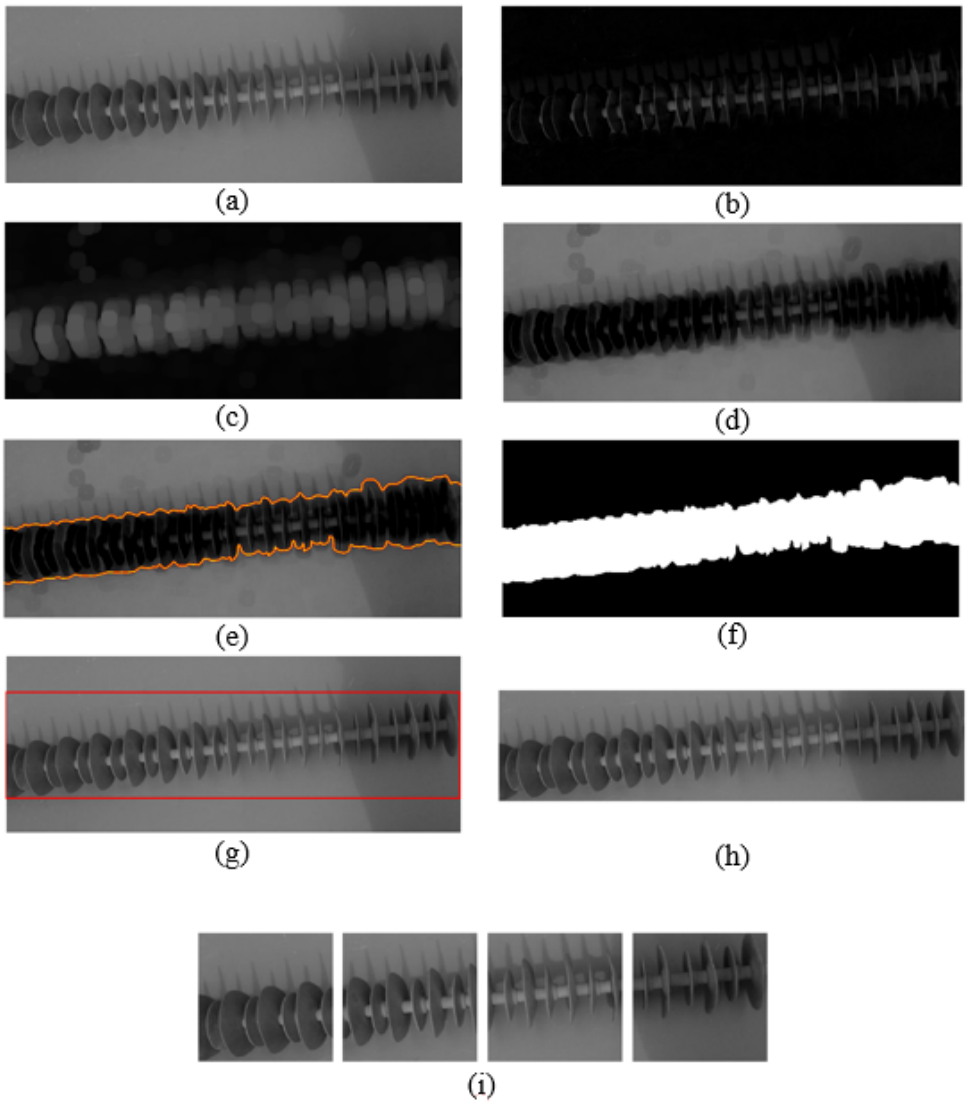


Figure 5.4: (a) Greyscale image, (b) Top-hat Filtering (c) Morphological dilation (d) Image subtraction (e) ACM (f) Segmented image (g) ROI (h) ROI extraction (i) Partitioned image. Best viewed in colour

5.3.3 Segmentation

Figure 5.4(e) shows the result of the application of ACM, where a line is drawn on the boundary of the insulators and the segmented image as shown in Figure 5.4(f). In

cases were there are some background pixels inside the insulator (foreground), post-processing is employed to fill up the holes.

5.3.4 Region of Interest Extraction

Figure 5.4(g) shows the identification of insulator ROI. This is obtained by extracting the coordinates, length and width that are used to define a minimum bounding box over the insulator from Figure 5.4(f), followed by the extraction of the insulator ROI from the image in Figure 5.4(a). The output is shown in Figure 5.4(h).

Due to the length of each insulator, it will be difficult to detect some smaller regions that are defectuous, therefore each insulator image is partitioned into smaller segments to enhance classification performance. Figure 5.4(i) shows an insulator image that has been partitioned.

5.3.5 Feature Extraction

SIFT is used for feature extraction in this chapter and its effectiveness has been compared with other feature methods used in Chapter 4 such as GLCM and LBP. The extracted features are fed into the KNN and SVM classifiers for insulator condition prediction.

5.3.6 Classification

Tables 5.1 - 5.8 show the condition of insulators based on 10-fold cross validation (CV) and test set implementation outputs respectively, while Tables 5.9 and 5.10 show the accuracy of GLCM, LBP and SIFT feature extraction methods with KNN and SVM classifiers. The performance analysis of the proposed method is measured using Equation 3.26.

Table 5.1: Confusion matrix for the cross-validation using SIFT with KNN

	Defectuous	Non-defectuous
Defectuous	200	0
Non-defectuous	0	200

Table 5.2: Confusion matrix of the test set using SIFT with KNN

	Defectuous	Non-defectuous
Defectuous	95	5
Non-defectuous	21	79

Table 5.3: Confusion matrix for the cross-validation using LBP with KNN

	Defectuous	Non-defectuous
Defectuous	200	0
Non-defectuous	0	200

Table 5.4: Confusion matrix of the test set using LBP with KNN

	Defectuous	Non-defectuous
Defectuous	94	6
Non-defectuous	24	76

Table 5.5: Confusion matrix for the cross-validation using GLCM with KNN

	Defectuous	Non-defectuous
Defectuous	190	10
Non-defectuous	10	190

Table 5.6: Confusion matrix of the test set using GLCM with KNN

	Defectuous	Non-defectuous
Defectuous	88	12
Non-defectuous	30	70

Table 5.7: Confusion matrix for the cross-validation using SIFT with SVM

	Defectuous	Non-defectuous
Defectuous	200	0
Non-defectuous	0	200

.

Table 5.8: Confusion matrix of the test set using SIFT with SVM

	Defectuous	Non-defectuous
Defectuous	93	7
Non-defectuous	18	82

Table 5.9: Accuracy of GLCM, LBP and SIFT using SVM

	10-fold validation ACC (%)	Testing ACC (%)
GLCM+SVM	94.8%	80.0%
LBP+SVM	100%	85.5%
SIFT+SVM	100%	87.5%

Table 5.10: Accuracy of GLCM, LBP and SIFT using KNN

	10-fold validation ACC (%)	Testing ACC (%)
GLCM+KNN	95%	79.0%
LBP+KNN	100%	85.0%
SIFT+KNN	100%	87.0%

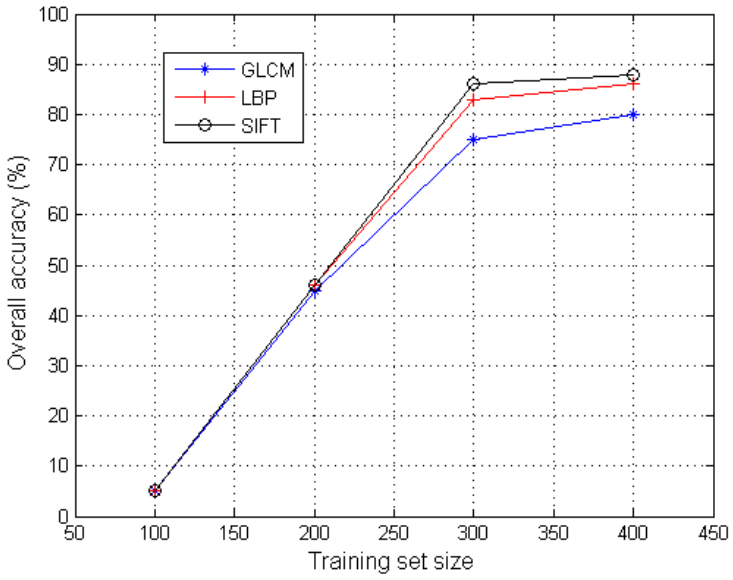


Figure 5.5: Accuracy in varying the training set size using SVM

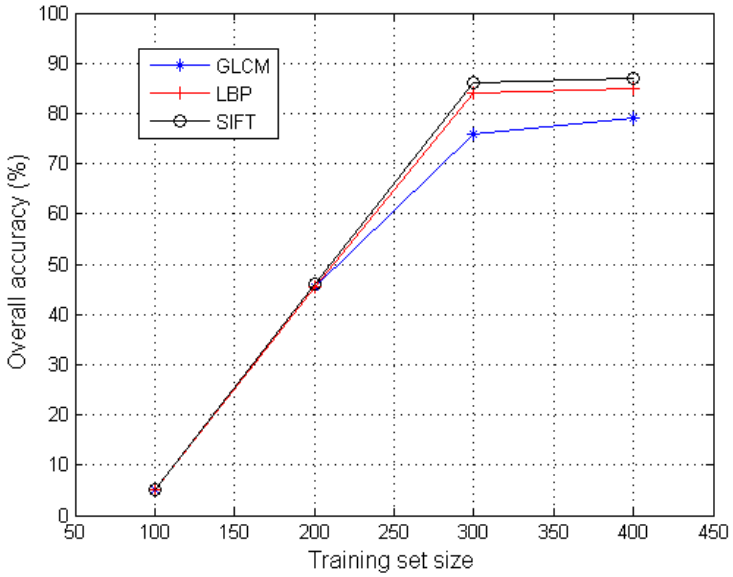


Figure 5.6: Accuracy in varying the training set size using KNN

Figures 5.5 and 5.6 show the variation of the training set size along side the test set size using SVM and KNN respectively. Increasing the training set size, increases the accuracy of insulator classification. When the training set is increased above 300, a gradual increase is noticed. From this point forward, there is a reduced error rate which may stay approximately the same within this region. The model performs best at this region.

5.4 Statistical Test

In order to validate the accuracy of the methods, such as LBP and SIFT using SVM, and LBP and SIFT using KNN, the McNemar's test is used. Let B be LBP with SVM, C be SIFT with SVM, D be LBP with KNN, and E be SIFT with KNN algorithm. Given the following number of cases, Table 5.11 is proposed.

- Number of insulators estimated defectuous for B and defectuous for C = a
- Number of insulators estimated non-defectuous for B and defectuous for C = b
- Number of insulators estimated defectuous for B and non-defectuous for C = c

- Number of insulators estimated non-defectuous for B and non-defectuous for C = d

Table 5.11: Contingency table

		B	
		Defectuous	Non-defectuous
C	Defectuous	a	b
	Non-defectuous	c	d

A contingency table is generated using algorithm B and C based on [154] and it is shown in Table 5.12.

Table 5.12: Contingency table. LBP and SIFT using SVM classifier

		B		
		Defectuous	Non-defectuous	
C	Defectuous	187	13	$t_1 = 200$
	Non-defectuous	41	159	$t_2 = 200$
		$t_3 = 228$	$t_4 = 172$	$T = 400$

The McNemar's test uses data from the two discordant entries b and c [153] from Table 5.11, where n is the total number of matched pairs ($n = a+b+c+d$). The McNemar's test is used to determine whether the null hypothesis H_o is accepted or rejected. The H_o states that there is no significant difference between the performances of the two algorithms and the alternative hypothesis H_a states that there is significant difference between the performances of both algorithms. The McNemar's test is computed using Equation 5.15 [154].

$$\chi^2 = \frac{(|b-c|-1)^2}{b+c} \tag{5.15}$$

where χ^2 is chi-squared distribution with one degree of freedom (DoF).

Therefore, the computed chi-square value (χ^2) using Equation 5.15 is 13.5. By convention, scientists often use P value of 0.05 (5%) for deviation of significant test. However, if it is greater, then the null hypothesis H_o is rejected. The P value of 0.05 for

1 Degree of Freedom in the chi-square (χ^2) distribution table is 3.841 [155]. Since the computed $\chi^2 > 3.841$, H_o is rejected. Therefore, it means that the two algorithms (B and C) are statistically different using the same test set. However, the same procedure is applied for algorithm D and E to have a computed χ^2 value of 19.45 (Table 5.13). The H_o is rejected. Then for the classification comparison of KNN and SVM, where the SIFT feature extraction method is used only due to its accuracy rate over LBP and GLCM, the computed χ^2 value is 8.65 (Table 5.14). The H_o is rejected, meaning there is a significant difference.

Table 5.13: Contingency table. LBP and SIFT using KNN classifier

		D		
		Defectuous	Non-defectuous	
E	Defectuous	189	11	$t_1 = 200$
	Non-defectuous	45	155	$t_2 = 200$
		$t_3 = 234$	$t_4 = 166$	$T = 400$

Table 5.14: Contingency table. SIFT using SVM and KNN classifiers

		C		
		Defectuous	Non-defectuous	
E	Defectuous	188	12	$t_1 = 200$
	Non-defectous	39	161	$t_2 = 200$
		$t_3 = 227$	$t_4 = 173$	$T = 400$

5.5 Conclusion

In this chapter, the classification of power-line insulator condition has been investigated and presented. The segmentation method was evaluated using manually created ground-truth by experts. Also the comparison of SIFT and LBP feature extraction method with KNN classifier was presented. It is shown that SIFT outperformed LBP with the use of KNN and SVM classifiers using the accuracy rates. It is also observed that variation in the training set size resulted in increased accuracy or reduced error rate. From the experiment, it is evident that the SIFT is robust, more stable and

less prone to noise. Also, based on the McNemar's test, it is observed that there is significant difference in their comparison.

Chapter 6

Hydrophobicity Classification of Insulator using Geometric Parameters

6.1 Introduction

This chapter discusses the condition of a polymeric insulator based on its hydrophobicity using geometric parameters. A polymeric insulator is high in hydrophobicity when newly installed and gradually degrades after a short time due to some factors, such as build-up of pollution layers, surface oxidation, and ageing of the polymeric insulator. These factors reduce the hydrophobicity of an insulator. Thus, in order to guarantee safety and optimal performance of the power line system, it is crucial to periodically estimate the status of insulators' hydrophobicity using an automated system such as a computer vision method instead of a manual method.

Also, this chapter describes the experimental procedure of water droplets on insulator surfaces, template preparation and its parameters, water droplet segmentation on the insulator and the extraction of geometric features. In addition, analysis and discussion of the experimental results are presented. One main approach to insulator hydrophobicity is the Swedish Transmission Research Institute (STRI) method.

6.2 The Swedish Transmission Research Institute Classification Method

The most accurate method of measurement is the use of stationary devices in the laboratory instead of field inspection which is usually limited to visual signs of degradation, such as cracks and de-colouration [134]. Hydrophobicity can be estimated when the surface of an insulator is sprayed with distilled water. This is the basis of the STRI classification method and it is based on a manual estimation of hydrophobicity classification (HC) for an insulator in the field. The STRI method is based on two key features: the receding angle and the area covered by the water (completely wet area).

Table 6.1: Reference table from STRI guide (HC1 to HC7), r = receding angle [9]

HC	Description
1	Only discrete droplets are formed. $\theta_r = 80^o$ or larger for the majority of droplets.
2	Only discrete droplets are formed. $50 < \theta_r < 80^o$ for the majority of droplets.
3	Only discrete droplets are formed. $20 < \theta_r < 50^o$. Usually they are no longer circular.
4	Both discrete droplets and wetted traces from the water runnels are observed (i.e $\theta_r = 0^o$). Completely wetted areas $< 2cm^2$. Together they cover $< 90\%$ of the tested area.
5	Some completely wetted areas $> 2cm^2$, which cover $< 90\%$ of the tested area.
6	Wetted areas cover $> 90\%$, i.e small, unwetted areas (spots/traces) are still observed.
7	Continuous water film over the whole tested area

The HC process commences with the use of a spray bottle to spray distilled water on an insulator surface and then the receding contact angle is determined. This is the angle between the water droplets and the insulator surface. An inspector then specifies the HC level of the insulator surface image. As a guide, the inspector has templates of insulator surface images with wet patterns for each HC level [157], [158], [159]. The

criteria for the HC are given in Table 6.1 and the typical STRI HCs from HC1 to HC6 are shown in Figure 6.1.

From Figure 6.1, HC1 to HC3 are hydrophobic since only water droplets are present, whereas HC4 to HC6 are partially hydrophilic since both wet areas and water droplets are present and HC7 is hydrophilic since the insulator surface is completely wet.

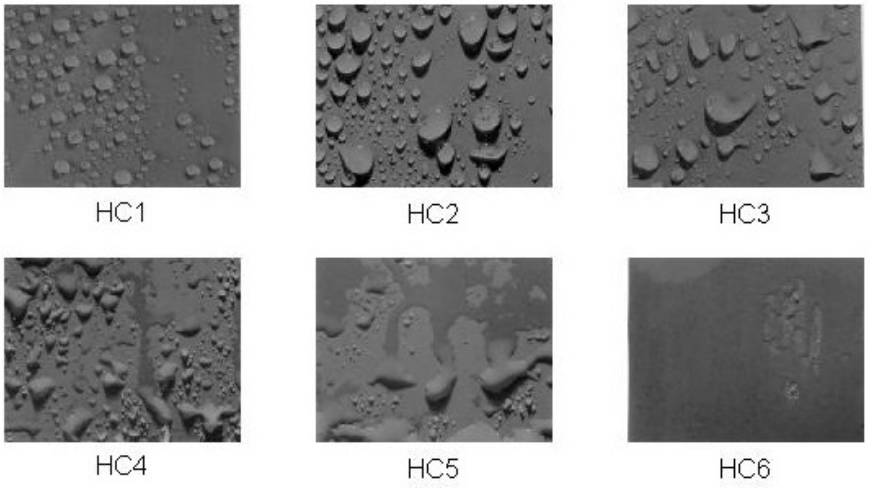


Figure 6.1: Reference images from STRI guide (HC1 to HC6) [9]

6.3 Experimental Procedure

6.3.1 Experiment

The experiment relies on the STRI guide [9]. A spray bottle is used to spray distilled water on a fixed insulator surface image. Then, a digital camera is used to capture images of water droplets on the insulator surface from a set distance as it gets to a stable state. The procedure above is applied on all considered image samples. The configuration parameter of the inclination angle is not provided in the STRI guide, which may result in different patterns of the water droplets under the same wetting conditions. However, this is not applicable to this experiment since the contact angle is not considered.

Firstly, captured images are converted from colour to greyscale, then the segmentation of water droplets on insulators and finally the classification of the insulator into different hydrophobicity levels. In this research work, the edge-based segmentation is

used instead of region-based segmentation because water is colourless and transparent, therefore has an effect upon the colour property, and secondly, pollutants on the insulator surface have an effect on the texture property [138]. Hence, the colour and texture properties of the water droplets are similar to the background which makes the edge-based approach is useful for indicating the water droplets on insulator surfaces.

In [9], the receding angle and the area covered are the two criteria to determine the HC levels of an insulator. HC1 to HC3 are estimated with the receding angle while HC4 to HC7 are estimated with the area covered by the water droplets. However, this experiment considers the shape (circularity) and the area covered by the water droplets since the receding angles cannot be computed from a digital image. Instead of the receding angle for HC1 to HC3, the circularity of the water droplets is introduced since it is stated in Table 6.1 that at HC3 level, "the majority of the water droplets are no longer circular". It is clearly stated in Table 6.1 that "only discrete droplets are formed" from HC1 to HC3, which means that no wet areas can be identified at these levels. In this experiment, the target object is the water droplet and can be characterized by colour, texture and geometric parameter [138]. The colour and texture parameters are hard to use in the classification of HC levels because water is colourless. Fortunately, the geometric parameter of the target object is independent of the water-sprayed pollutants on the insulator surface and the natural pollutants on the insulator surface: therefore the geometric parameters are useful for HC.

6.3.2 Template Preparation and its Parameters

In our experiments, images (see Figure 6.1) of size 330 x 256 pixels from the STRI guide are used. The edges of individual water droplets in images of different HC levels are well traced out by experts (see Figure 6.2) in order to have standard templates with a binary image to estimate and extract circularity threshold (TC_D), area cover threshold (TAC) and the threshold of minimum area of wet trace (TAC_W). (TC_D) is the maximum circularity threshold of water droplets extracted for each HC class from HC1 to HC3 in Figure 6.2. TA_D is the area threshold of water droplets extracted for each HC class from HC1 to HC7 in Figure 6.2. TAC is the area cover threshold of water droplets extracted for each HC class from HC1 to HC7 in Figure 6.2. TAC_W is the threshold of minimum area of wet trace from a range greater than maximum area

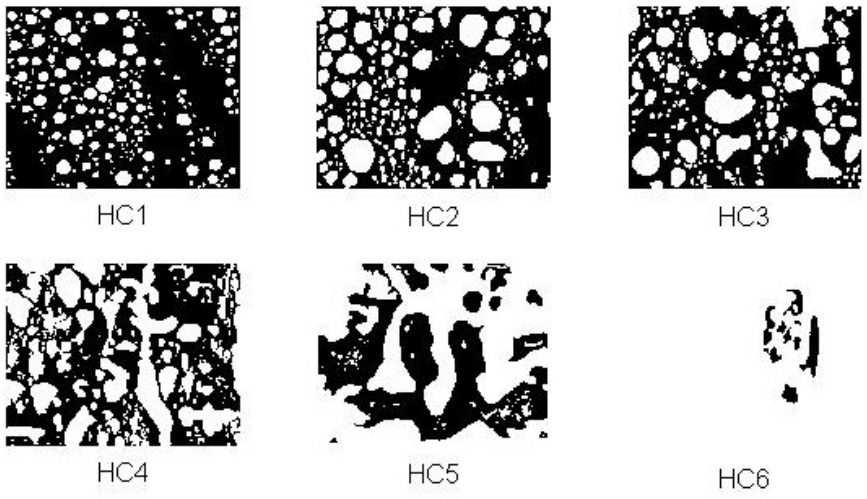


Figure 6.2: Reference images from STRI guide (HC1 to HC6), binarized by expert

Figure 6.3: Magnified images of HC1, HC3 and HC4 in Figure 2 with few water droplets

of HC3 and less than maximum area of HC4 in Figure 6.2. The extracted and defined threshold range serves as a reference to classify other images (test images) with their extracted circularity C_D, area A_D, area cover AC and area of wet trace AC_W values into different HC levels.

Figure 6.3 shows a set of magnified images of HC1, HC3 and HC4 in Figure 6.2 with some water droplets in HC1 and maximum water droplets in HC3 and HC4 for the purpose of template parameter explanation. Let D represent the connected components of the pixels; A_D represents the area of individual water droplets $(1, 2, 3...10)$ in HC1 of Figure 6.3, therefore the area cover threshold TAC is $(TAC = A_{D(1)} + A_{D(2)} + A_{D(3)} + ... + A_{D(10)})$. In Table 6.1, wet traces are observed in HC4, therefore TAC_W is a threshold area greater than the maximum area of HC3, $A_{D(3)}$, but less than the

maximum area of HC4, $A_{D(4)}$ in Figure 6.3. In other words, this can be defined as $(A_{D(3)} > TAC_W < A_{D(4)})$. TAC_W is set as a minimum threshold for classification from HC4 to HC7. Considering HC1 in Figure 6.3, let Mj represent the major axis and Mn represent the minor axis of each water droplet $(1, 2, 3...10)$. The circularity threshold TC_D is the maximum difference between Mj and Mn in individual water droplets in an image of each HC level. Considering HC1 in Figure 6.3, $TC_{D(HC1)}$ is $max[(Mj_1 - Mn_1), (Mj_2 - Mn_2), ..., (Mj_{10} - Mn_{10})]$. This is applicable to HC2 and HC3 where circularity plays a vital role.

6.4 Water Droplet Segmentation

6.4.1 Image Pre-processing

Captured images are mostly in RGB form. The RGB model is often not suitable in many computer vision applications, hence it is often converted into other models such as the greyscale. In this work, greyscale images are used. A series of morphological operations [27], [28], such as dilation, erosion and closing operation; are used to remove noise and further enhance the images.

6.4.2 Water Droplet Segmentation using Geodesic Active Contour

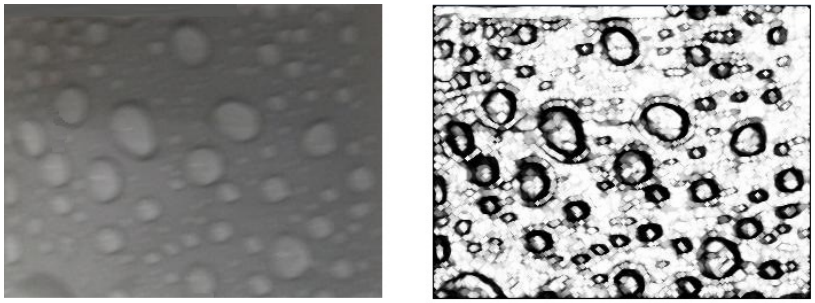
The model presented in this research work for water droplet segmentation on insulator surface images is based on [47], [21], [160]. Geodesic Active Contour (GAC) is expressed as:

$$\frac{\partial \Gamma(t)}{\partial t} = v(m)\kappa\overline{R} - (\nabla v \bullet \overline{R})\overline{R} \tag{6.1}$$

where $\overline{R}$ is the unit normal vector of the curve Γ, κ is the Euclidean curvature, $v(m)$ is the stopping function and ∇v is the gradient of the potential edge indication function v expressed as:

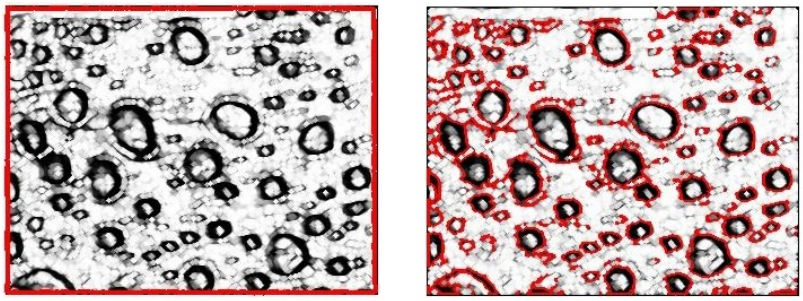
$$v = \frac{1}{1 + |\nabla(\theta_\sigma \star m)|^c} \tag{6.2}$$

where m is the image given, θ_σ is the Gaussian filter, $(\theta_\sigma \star m)$ is the smoothed version of m and c is a constant which is 2 in this experiments.



(a) (b)

Figure 6.4: Image transformation (a) Original image and (b) Output of edge indicator



(a) (b)

Figure 6.5: Geodesic transformation (a) Initial curve at the border of insulator image (b) Final curve at the edges of the water droplets (best viewed in colour)

Equation 6.2 is applied to the original image in greyscale, m, in Figure 6.4(a). m is smoothed with a Gaussian filter θ_σ. Then, very small values of v define the local edges (see image transformation in Figure 6.4(b) for the curve to travel through. Then, Equation 6.1 is applied onto Figure 6.5(a) with an initial curve Γ at the boundary of image, and $v(m)$ provides a constant speed for the curve evolution in order to accelerate the convergence. The value of ∇v pushes the curve to the edges of the water droplets and stops when $v(m) = 0$ at the boundary of the water droplets (see Figure 6.5(b)).

Algorithm 6 presents the steps involved in water droplet segmentation on polymeric insulators based on GAC.

Algorithm 6 Water Droplets Segmentation

Require: m ▷ Source Image
Ensure: $segIm$ ▷ Segmented Image
1: **Convert** the RGB image m into grayscale and Save as gm
2: **Apply** Equation 6.2 to gm and Save as gm
3: **Define** a structuring element S of size 2 by 3
4: **Apply** S to dilate gm, and Save as d
5: **Apply** Equation 6.1 to d and Save as $Igac$
6: **Apply** S to erode $Igac$, and Save as e
7: **Apply** morphological closing operation to e
8: **Save** the segmented image e as $segIm$
9: **End**

6.5 Extraction of Geometric Features

The segmented image, $segIm$, of water droplets on the insulator, contains pixel values 0 and 1 representing background and foreground respectively. In other words, $segIm$ is defined as:

$$segIm(r,c) = \begin{cases} 0 & \text{the pixel belongs to the background} \\ 1 & \text{the pixel belongs to the foreground} \end{cases}$$

However, the interest is based on geometric parameters which are shape and area of water droplet. These parameters are used in estimating HC level.

6.5.1 Area of Individual Water Droplet Region

To obtain the area of individual water droplet region, it is necessary to transverse through each pixel in the insulator surface image from left to right and top to bottom until the last location or position in the image is reached [161]. When moving across pixels, if any pixel that is equal to the foreground value is encountered, it checks its neighbouring pixels using 8-connected component and label this connected component. In this work, an 8-connected component is used instead of a 4-connected component because the 8-connected component considers the adjacent and diagonals of the parent pixel, thereby having a better description of the water droplet region for classification, unlike the 4-connected component which considers the adjacent pixels only.

The area of individual water droplet region, A_D, is the sum of the pixels, D_p, in the connected component representing this droplet. In other words, the area is defined as

$$A_D = \sum D_p \tag{6.3}$$

6.5.2 Area Cover

The area cover is one of the important characteristic parameters used in the STRI guide (see Table 6.1) for deciding hydrophobicity of insulators and represents the overall hydrophobicity of the insulator surface image. The area cover can also be referred to as the density of water distribution on an insulator surface image. The area cover (AC) is the summation of all the individual water droplets in an insulator surface image. The area cover of a segmented insulator surface image is defined as:

$$AC = \sum_{i=1}^{n} |A_{Di}| \tag{6.4}$$

where AC is the area cover of all the individual water droplets, and A_D is the area of individual droplets region in the segmented image $segIm$.

AC_W is the summation of all the individual areas of water droplets greater than the threshold set as TAC_W. TAC_W is greater than the maximum area in HC3 and less than maximum area of HC4.

6.5.3 Circularity of Individual Water Droplet

Circularity is a measure of the extent to which the water droplet is similar to a circle [94]. There are different measures to define the shape of an object. In this work, the circularity of the water droplet is based on the ellipse major and minor axis lengths (in pixels). The major and minor axes pass through the centre of mass of the water droplet. The minor axis is perpendicular to the major axis. Water droplets are circular when the difference between the major and minor axis length is equal and tends to be elliptical or irregular when the major axis length increases farther away from the minor axis length. The circularity is defined as:

$$C_D = max[(M_{j(1)} - M_{n(1)}), (M_{j(2)} - M_{n(2)}), ..., (M_{j(n)} - M_{n(n)})] \tag{6.5}$$

where C_D is the circularity of the individual water droplet region, M_j is the major axis, M_n is the minor axis and $1, 2, ..., n$ are individual droplets in $segIm$.

6.5.4 Criteria for Evaluating HCs

Table 6.2 shows the criteria for evaluating HC. From the HC1 to HC3 evaluation, both C_D and AC are used since it is stated in Table 6.1 that the water droplets are circular from HC1 to HC2 until HC3 where the majority are no longer circular. Circularity is used in HC3 since it is the only way to distinguish it from other classes. Also from HC1 to HC3 classes, there is no area greater than AC_W. The criteria for evaluating HC4 to HC6 in Table 6.2 are the same as in Table 6.1. In these classes, the circularity is of no use. Instead AC and AC_W are used. HC4 is based on AC and the AC_W. AC covers less than 90% of the tested area and AC_W is the summation of areas greater than the threshold set as TAC_W. For HC5, AC only covers less than 90% of the tested area. For HC6, AC only covers more than 90% of the tested area. Then, AC is equal to 100% of the tested area, meaning that the entire surface of the tested area is wet. If no water droplet is present or identified on the tested insulator surface image, it is identified to be in HC7 class. With this, a decision-making classifier using an "if-else" statement is used for the evaluation of the HC levels.

Table 6.2: Criteria for evaluating HCs

HCs	Description
HC1	C_D and AC
HC2	C_D and AC
HC3	C_D and AC
HC4	AC and AC_W covers $< 90\%$ of the tested area
HC5	AC and only AC_W covers $< 90\%$ of the tested area
HC6	AC and only AC_W covers $> 90\%$ of the tested area
HC7	$AC = 100\%$ of the tested area

6.5.5 Algorithm for Hydrophobicity Classification of Insulators

Algorithm 7 gives the steps involved in hydrophobicity classification of polymeric insulators based on geometric parameters. Let m denotes maximum value of the threshold

Algorithm 7 Hydrophobicity Classification of Insulators

Require: $segIm$ ▷ Segmented Image
Ensure: $HCLevel$ ▷ HC Level

1: $(D_1, D_2, ..., D_{ncc}) \longleftarrow$ Find connected component $(SegIm)$
2: **for** $i = 1$ **to** cc **do**
3: **Compute** A_{Di} using Equation 6.3
4: **Compute** AC using Equation 6.4
5: **Find** and **Sum** all $A_D > TAC_W$ and save as AC_W
6: **Compute** C_D using Equation 6.5
7: **end for**
8: **if** $(C_D < TC_D)$ **then**
9: **if** $(C_D \leq TC_{D(2)m})$ **then**
10: **if** $(AC > 0)$ &&$(AC \leq TAC_{(1)m})$ **then**
11: **Display** HC1
12: **else**
13: **Display** HC2
14: **end if**
15: **end if**
16: **if** $(C_D > TC_{D(2)m})$ &&$(C_D \leq TC_{D(3)m})$ **then**
17: **if** $(AC > 0)$ &&$(AC \leq TAC_{(3)m})$ **then**
18: **Display** HC3
19: **end if**
20: **end if**
21: **else**
22: **if** $(AC_W > 0)$ &&$(AC_W < T90)$ **then**
23: **if** $((AC > TAC_{(3)m})$ &&$(AC \leq TAC_{(4)m}))$ || $((AC_W > TAC_W)$ &&$(AC_W \leq TAC_{W(4)m}))$ **then**
24: **Display** HC4
25: **else**
26: **if** $((AC > TAC_{(4)m})$ && $(AC \leq TAC_{(5)m}))$ || $((AC_W > TAC_{W(4)m})$ &&$(AC_W < T90))$ **then**
27: **Display** HC5
28: **end if**
29: **if** $(AC_W > T90)$ &&$(AC_W < T100)$ **then**
30: **Display** HC6
31: **end if**
32: **if** $(AC_W == T100)$ **then**
33: **Display** HC7
34: **end if**
35: **end if**
36: **end if**
37: **end if**
38: **End**

parameters, real numbers $1, 2, 3...7$ represent HC1, HC2,...HC7 respectively, $T90$ and $T100$ represent 90% and 100% of the tested images respectively.

6.6 Experimental Results and Discussion

6.6.1 Dataset

A dataset of 90 polymeric insulator surface images are used in this experiment. From the literature survey, there is presently no publicly available dataset for insulator studies on hydrophobicity. Hence, the experiment on insulator hydrophobicity is based on the dataset of this experiment.

6.6.2 Performance Analysis

The performance analysis of the proposed method is measured using some statistical criteria and confusion matrix for visualization. The statistical criteria are sensitivity (Se), specificity (Sp), precision (Pr), accuracy (Ac) and F1-Score (Fs) [149, 150]. These statistical criteria are defined in Chapter 3 as Se (Equation 3.23), Sp (Equation 3.24), Pr (Equation 3.25), Ac (Equation 3.26) and Fs (Equation 3.27).

6.6.3 Results Analysis

In this experiment, the expansion weight, time step and number of iteration for the GAC are set to -1, 2 and 500 respectively. The negative sign for the expansion weight denotes the case of curve deformation.

Figure 6.6 shows the comparison of the proposed method with [10] and [135] methods. In addition, only the performance of the proposed method and the Dong et al. [10] method are qualitatively evaluated. The Dong et al. [10] method of segmentation is based on the combination of two methods namely adaptive-threshold and canny operator. In the Liang et al. [135] method, Canny edge detector is used to detect edges separately in the saturation and brightness channels of the HSV (Hue, Saturation and Brightness). Thereafter, a threshold is applied to eliminate noise, then edges in both channels are combined.

In Figure 6.6, the edges of the water droplets in column c of the proposed method are better outlined as the ground-truth and original images than in columns d and

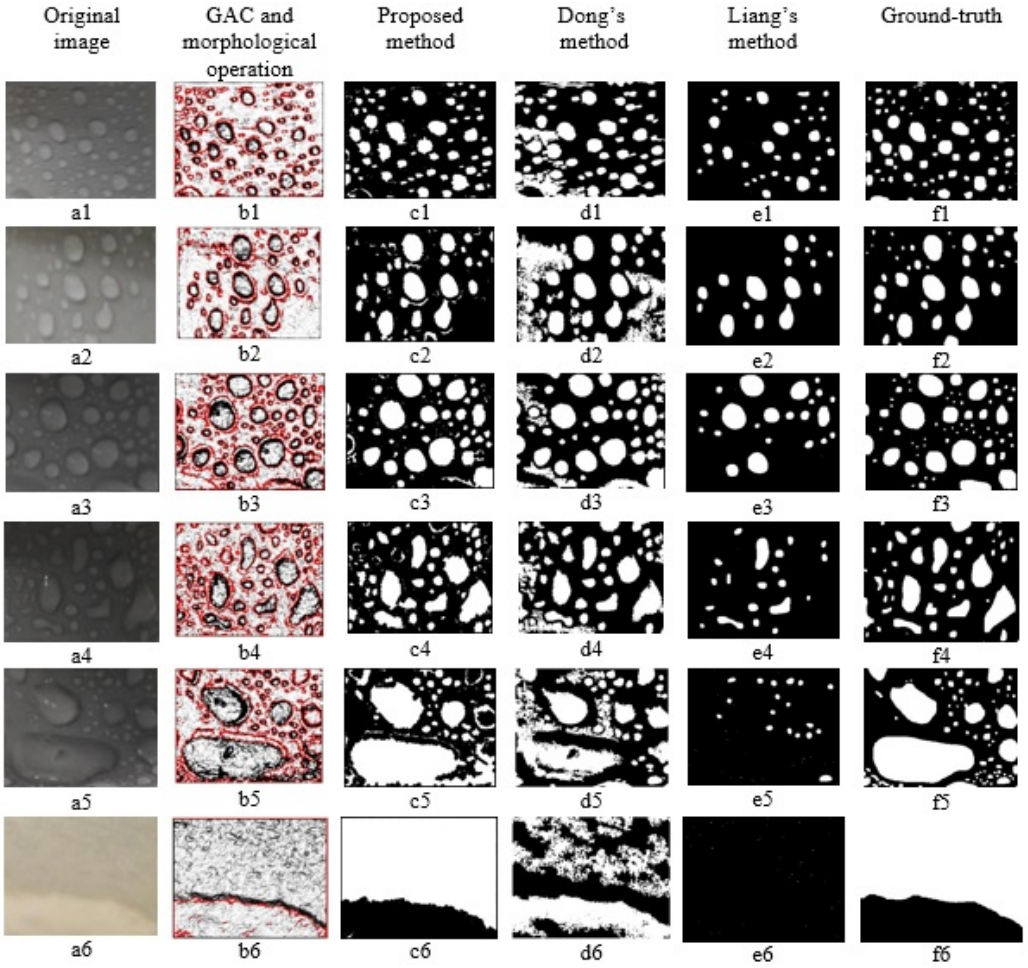


Figure 6.6: Result of water droplet segmentation on insulator. Best viewed in colour

e. The Liang et al. [135] method has poor performance because of the sensitivity of Canny operator to noise. Poor performance is observed in $e5$ and $e6$ because there are no closed edges in the water droplets. Few water droplets are observed in $e1$ and $e4$ because of the method used to eliminate edges greater than a certain threshold. This method is appropriate when there is continuity in the edges of the water droplets such as $e1$ to $e5$. The segmentation method for the Dong et al. [10] method is not robust due to the fact that the valley in the bimodal histogram vanishes in noisy images, and as a result making it hard to compute and have a well-defined threshold parameter. Therefore, it limits this method to a great extent. There is reduced performance in segmentation on $d1$ and $d6$ due to the variation in the texture of the original image.

Also, there is reduced performance in $c1$ to $c6$ because of edge discontinuity of all water droplets at the borders of each polymeric insulator surface image (cut-out portions of the water droplets at the image borders).

Table 6.3: Performance analysis on water droplet segmentation, with values of sensitivity, specificity, precision, accuracy and f1-score in percentage (%). P = Proposed method and D = Dong et al. [10] method

	Sensitivity		**Specificity**		**Precision**		**Accuracy**		**F1-score**	
	P	D	P	D	P	D	P	D	P	D
Average	83.84	70.74	79.25	71.66	76.15	68.05	82.26	70.22	79.15	67.42

Furthermore, the segmentation approach of both the proposed method and the Dong et al. [10] method using the ground-truth of the insulator surface images, is quantitatively evaluated with the following statistical criteria (see Equations 3.23, 3.24, 3.25, 3.26 and 3.27 in chapter 3): Se, Sp, Pr, Ac and Fs. Table 6.3 shows the average of the performance metric on water droplet segmentation. The proposed method has a higher values over Dong's method with the following performance metrics Se, Sp, Pr, Ac and Fs. Considering the accuracy (Ac) of the statistical criteria in Table 6.3, the proposed method has an accuracy (Ac) of 82.26% to Dong's method with an accuracy (Ac) of 70.22%. This shows that the proposed method is an improvement of water droplet segmentation on insulator surfaces images.

Table 6.4: Confusion matrix showing the performance of the proposed method on hydrophobicity

	HC1	HC2	HC3	HC4	HC5	HC6	HC7
HC1	9	2	0	0	0	0	0
HC2	1	21	2	0	0	0	0
HC3	0	0	9	1	0	0	0
HC4	0	0	3	10	1	0	0
HC5	0	0	0	2	21	2	0
HC6	0	0	0	0	0	2	2
HC7	0	0	0	0	0	0	2

Table 6.5: Confusion matrix showing the performance of the Dong et al. [10] method on hydrophobicity

	HC1	HC2	HC3	HC4	HC5	HC6	HC7
HC1	7	2	2	0	0	0	0
HC2	1	18	4	1	0	0	0
HC3	0	0	8	1	1	0	0
HC4	0	0	1	12	1	0	0
HC5	0	0	0	1	23	1	0
HC6	0	0	0	0	1	2	1
HC7	0	0	0	0	0	0	2

Tables 6.4 and 6.5 show the confusion matrices performance of the proposed method and the Dong et al. [10] method respectively. Tables 6.6 and 6.7 show the performance analysis of HC level based on Se, Sp, Pr, Ac and Fs. Tables 6.4 and 6.5 are multi-class classification, the one against all approach is used to estimate the True Positive TP_v, True Negative TN_v, False Positive FP_v and False Negative FN_v. The seven diagonal cells show the number of correct classifications and the ones outside the diagonal cells are misclassified. For example in Table 6.4, the diagonal cells value are $(9, 21, 9, 10, 21, 2, 2)$ and considering these values as a pivot point, all values along the horizontal axis of the pivot point, excluding the pivot point is considered as misclassified. The value 21 from the diagonal cell values in Table 6.4 is the number of correctly classified images into HC2 out of a total number of 24 images as HC2 while a total number of 1 and 2 images are misclassified into HC1 and HC3 respectively, with total number of misclassified images as 3. Considering HC2 in Table 6.5, the number of images that are correctly classified into HC2 is 18 out of a total of 24 images as HC2 while a total number of 1 and 4 images are misclassified into HC1 and HC3 respectively, with total number of misclassified images as 5. This misclassification reduces the accuracy of the performance. In Tables 6.6 and 6.7, the accuracy value for the proposed method and Dong's methods are 93.67% and 90.00% for HC2 classification respectively. There is a reduced percentage value for HC2 classification in Dong's method over the proposed method because more images are misclassified with Dong's method. The same principle is applicable to other HC levels. However, there is an accuracy drop due to misclassification from one HC level to another HC level resulting in a false negative. It

is observed that the proposed method has more images classified into HC1, HC2, HC3 and less number of images into HC4, and HC5 over Dong's method. The same number of images are classified into HC6 and HC7 for both methods. Tables 6.6 and 6.7 show the overall average accuracy of 82.22% and 80.00% for the proposed method and the Dong et al. [10] methods respectively.

Table 6.6: Performance analysis of proposed method on hydrophobicity. All parameters are in percentages (%)

	HC1	HC2	HC3	HC4	HC5	HC6	HC7	Average	Overall Average
Ac (%)	96.10	93.67	92.50	91.36	93.67	94.87	97.37	94.22	82.22
Pr (%)	90.00	91.30	64.29	76.92	95.45	50.00	50.00	74.00	
Se (%)	81.82	87.50	90.00	71.43	84.00	50.00	100.00	80.68	
Sp (%)	98.48	96.36	92.86	95.52	98.15	97.30	97.30	96.57	
Fs (%)	85.71	89.36	75.00	74.07	89.36	50.00	66.67	75.74	

Table 6.7: Performance Analysis of Dong et al. [10] method on hydrophobicity. All parameters are in percentages (%)

	HC1	HC2	HC3	HC4	HC5	HC6	HC7	Average	Overall Average
Ac (%)	93.51	90.00	88.89	93.51	93.51	96.00	98.63	93.43	80.00
Pr (%)	87.50	90.00	53.33	80.00	88.46	66.67	66.67	76.09	
Se (%)	63.64	75.00	80.00	85.71	92.00	50.00	100.00	78.05	
Sp (%)	98.48	96.43	90.14	95.24	94.23	98.59	98.59	95.96	
Fs (%)	73.68	81.82	64.00	82.76	90.20	57.14	80.00	75.66	

Furthermore, the accuracy of HC1 to HC7 are used to investigate the significance of the difference over both methods of hydrophobicity classification. The mean and standard deviation of the accuracy for the statistical criteria are computed and used to plot the bar graph shown in Figure 6.7. The blue bars (standard error bars) at the middle of each bar plot (mean bars) represent the standard deviation. It is observed that both standard error bars overlap, which means that the difference between the two means of accuracy is not statistically significant.

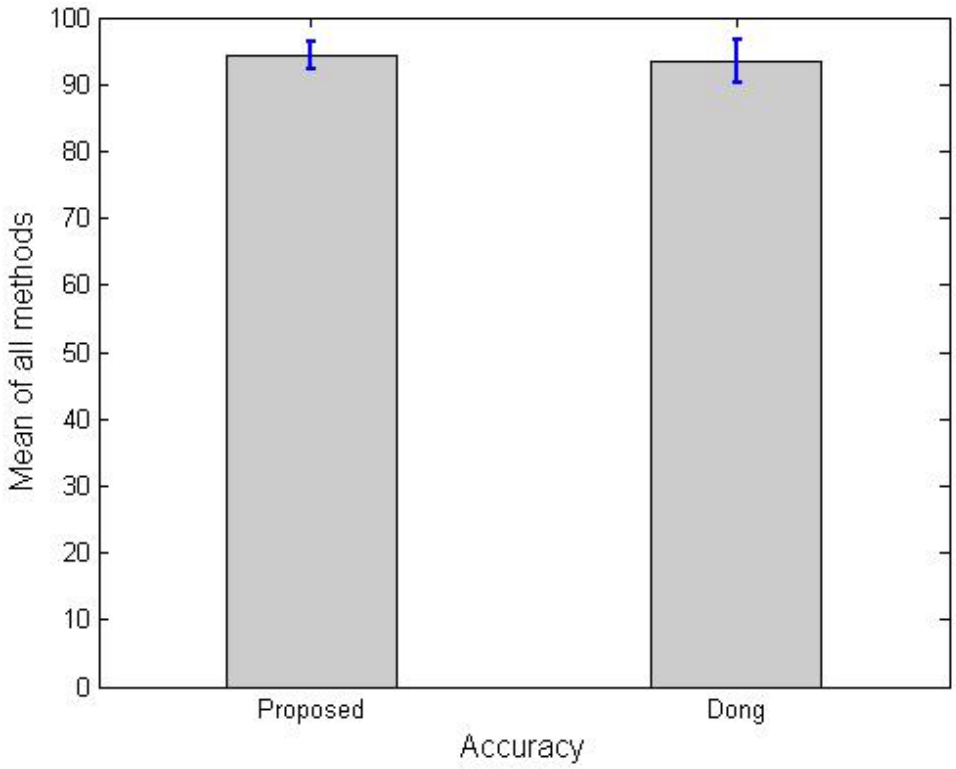


Figure 6.7: Mean and standard deviation of the proposed method and Dong et al. [10] method of hydrophobicity classification . Best viewed in colour

The area of the water droplets is affected by under- and over-segmentation. This means that the area is either too large or too small, resulting in misclassification of HC levels. Under-segmentation and over-segmentation are experienced due to poorly defined edges of water droplets. For uniform and non-uniform illumination images, the proposed method has better segmentation results over the Dong et al. [10] method when tested on our dataset (see Figure 6.6). This is because water is transparent and colourless and therefore there is only a slight difference between water droplets and background except at the edges. Also, the performance is reduced in some cases where part of the water droplet is outside the entire insulator surface image boundaries, such as $c1$ to $c5$ in Figure 6.6 .

Even after segmentation, there is good performance in the proposed method and the Dong et al. [10] method because the classification method also considers the density of

the water droplet region. This means that some areas with over- or under-segmentation in the polymeric insulator surface image can be compensated by another area where there is under- or over-segmentation, or even areas where there is an illumination problem, especially in the case of the Dong et al. [10] method (see Figure 6.6) $d1$ to $d6$ therefore have about same total cover area.

Liang and collaborators' [135] method experiences more over-segmentation because of its method of application, therefore have a reduced performance compared to other methods.

When water droplets are touching or are very close to each other, it becomes difficult to segment them. This affects the circularity of the water droplet, which reduces the performance and misclassification occurs.

6.7 Conclusion

Hydrophobicity is a way to monitor and inspect insulators before failure. Computer vision is adopted to replace manual inspection of HC. GAC has been used to segment the water droplets from the insulator surface image, which is very important in the classification stage. Then, different geometric parameters such as the area, area cover and circularity have been extracted for classification. Experimental results show that the proposed method in this thesis can effectively perform the HC of an insulator. A model to segment water droplets on an insulator and classify their state into different HC levels has been proposed. The main limitation of the proposed model is when edges are very weak, not regular or discontinued.

Chapter 7

Conclusion and Future Work

7.1 Thesis Conclusion

In this thesis, a frame work for the modelling and implementation of power-line insulator condition has been presented based on the experimental dataset since there is presently no publicly available dataset for insulator studies.

Power-line insulators are used in dangerous environments with a high potential difference between the lines generating an electric and magnetic field. The challenges of on-site inspection of insulator condition using the traditional or manual method is very dangerous and unsafe to humans and becomes an impractical task when inspecting long lines spanning through a long distance with difficult terrain. Therefore the manual method is not able to produced satisfying results. In order to overcome this problem, robust automated systems have been developed that can be used to identify and analyse insulator condition safely, speedily and accurately. Insulator condition described in this thesis is the structural inspection of defects on insulators such as cracks and broken parts and the hydrophobicity of an insulator under wet conditions.

An automated model has been developed for the physical condition of power-line insulators and composes of segmentation, feature extraction and classification. The algorithm for segmenting and extracting an insulator region of interest has been developed. The algorithm is used to segment the insulator, then to identify the insulator ROI, followed by the extraction of the insulator ROI. Segmentation plays a vital role in order to be able to extract features from the insulator for training purposes. The segmentation of an insulator was carried out using ACM. The mathematical imple-

mentation of the proposed ACM is accomplished using the level set method. ACM presents contours as a level of a topological function, that can merge or split, providing a good flexibility in the use of ACMs. ACM considers the uniformity of sub-regions, and therefore they can produce more reasonable segmentation results than traditional segmentation methods, and consequently improve the final results of image analysis. The effectiveness was evaluated using an accuracy rate with a higher accuracy value for the proposed method of segmentation over the existing method. T-test was used for accuracy and to further determine the statistical significance of the proposed method in this thesis over the existing method of segmentation. It was observed that there is no statistical significant difference. However, this actually depends on the size of the dataset used in an experiment.

Different feature extraction algorithms such as GLCM, LBP and SIFT were used and comparisons were carried out on the performance of these feature extraction methods in combination with different classifiers, KNN and SVM. It was observed that LBP outperformed GLCM with the use of the same SVM classifier considering the accuracy rate. It is evident that the LBP combined with SVM is robust, more stable and less prone to noise. It was also observed that variation in the training set size resulted in increased accuracy or reduced error rate. Furthermore, the McNemar's test, showed that there is a significant difference between LBP and GLCM using SVM classifier. A comparison of SIFT and LBP feature extraction methods was explored. SIFT outperformed LBP with the use of the same KNN classifier using the accuracy rate. It was also observed that there is a statistically significant difference between both algorithms using the McNemar's test.

In addition, an automated system has been developed for the hydrophobicity of insulators under wet conditions. The model is used for segmenting water droplets on the insulator surface image, followed by the extraction of geometric parameters such as the area of individual water droplet region, area cover and the circularity of individual water droplets, for the classification of an insulator into different hydrophobicity levels. Geodesic active contour was used for segmentation of the water droplets on insulator surface images because the texture properties of the water droplets are similar to the background which makes the edge-based segmentation useful over region-based approach for indicating the water droplets on insulator surfaces. The experiment showed

that the proposed model effectively performed the classification of insulators into different hydrophobicity level.

7.2 Limitation of the Automated System/Computer vision Method

The following are some limitations identified during the course of this study:

- The dataset for the work is restricted to a specific type of insulator known as the polymer insulators.
- The kind of defect defined in the dataset can be categorized as physical defects. Therefore, the identification of internal defects on insulators will not be possible using this method.
- Cracks and broken parts that are so tiny and almost unnoticeable affect the classification performance.
- The geometric parameter used as a template is based on images in the STRI guide and the template parameter can vary from one institution to another due to different experts defining the ground-truth.
- Very weak edges, or non-regular or discontinued edges affect water droplets segmentation.

7.3 Recommendation for Future Work

The following is a list of recommendations for future work:

- There is a need to create and make publicly available an insulator dataset in order to serve as a bench mark and for the purpose of comparison.
- Further investigation on other feature extraction and classification methods should be implemented.
- A liquid different from the insulator colour should be used to spray the insulator for hydrophobicity classification. This will help to define the edges properly. Also, a region-based method could therefore be used for segmentation.

- Internal defects in insulators should be considered. The use of computer vision combined with infra-red techniques for automated systems could be employed.

References

[1] "Metro power transmission," http://metropower.ca/services/transmission/, Accessed: March 5, 2017. xiii, 3

[2] "Mace technologies," http://www.macetech.co.za/HAWKDESC1.htm, Accessed: March 5, 2017. xiii, 4

[3] "Drone life," http://dronelife.com/2015/12/09/drones-power-bright-revolution-in-utility-inspection/, Accessed: March 5, 2017. xiii, 5

[4] "UKZN power line robot," http://www.ukzn.ac.za/news/2015/05/14/ukzn-robot-shortlisted-for-uk-energy-innovation-award, Accessed: March 5, 2017. xiii, 5

[5] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002. xiv, 22, 23, 24, 36, 64, 65, 66

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. xiv, xvii, 25, 26, 91

[7] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on image processing*, vol. 10, no. 2, pp. 266–277, 2001. xv, 16, 18, 19, 43, 44, 45, 46

[8] Z. Xinye, J. An, and F. Chen, "A method of insulator fault detection from airborne images," in *Second WRI Global Congress on Intelligent Systems*, vol. 2.

IEEE, 2010, pp. 200–203. xv, xviii, 21, 35, 36, 52, 53, 54, 55, 56, 57, 58, 77, 79, 80

[9] STRI, "Guide 1, 92/1: Hydrophobicity classification guide," *Swedish Transmission Research Institute*, pp. 1–5, 1992. xvii, xix, 105, 106, 107

[10] Z. Dong, Y. Fang, X. Wang, Y. Zhao, and Q. Wang, "Hydrophobicity classification of polymeric insulators based on embedded methods," *Materials Research*, vol. 18, no. 1, pp. 127–137, 2015. xvii, xix, 17, 115, 116, 117, 118, 119, 120, 121

[11] S. M. Kaplan, "Electric power transmission: background and policy issues," *US Congressional Research Service*, vol. 14, pp. 4–5, 2009. 1

[12] Z. Xinye, J. An, and F. Chen, "A simple method of tempered glass insulator recognition from airborne image," in *International Conference on Optoelectronics and Image Processing (ICOIP)*, vol. 1. IEEE, 2010, pp. 127–130. 1, 17, 35

[13] J. Katrasnik, F. Pernus, and B. Likar, "A climbing-flying robot for power line inspection," in *Proceedings of the IEEE Conference on Robotics, Automation and Mechatronics*. IEEE, 2008, pp. 95–110. 3, 4

[14] P. S. Prasad and B. P. Rao, "LBP-HF features and machine learning applied for automated monitoring of insulators for overhead power distribution lines," in *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE, 2016, pp. 808–812. 4, 36

[15] M. J. B. Reddy, D. K. Mohanta, and K. B. Chandra, "Condition monitoring of 11kV distribution system insulators incorporating complex imagery using combined DOST-SVM approach," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 20, no. 2, pp. 664–674, 2013. 6, 36

[16] EPRI, "Field guide - visual inspection of NCI," Electric Power Research Institute (EPRI), Palo Alto, CA, Tech. Rep. 1008739, 2004. 10, 11

[17] R. S. Gorur and T. Orbeck, "Surface dielectric behavior of polymeric insulation under HV outdoor conditions," *IEEE Transactions on Electrical Insulation*, vol. 26, no. 5, pp. 1064–1072, 1991. 11

[18] M. J. Owen, "Surface properties of silicone high voltage insulators," in *Science and Technology of Polymers and Advanced Materials*. Springer, 1998, pp. 99–106. 11

[19] J. Kim, "Hydrophobicity loss and recovery of silicone HV insulation," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 6, no. 5, pp. 695–702, 1999. 11

[20] IEC62072, "Guidance on the measurement of wettability of insulator surfaces," IEC Technical Specification TS 62072-2003, Tech. Rep., 2003. 11

[21] U. Iruansi, J. R. Tapamo, and I. E. Davidson, "An active contour approach to water droplets segmentation from insulators," in *Proceedings of the IEEE International Conference on Industrial Technology (ICIT2016)*. IEEE, 2016, pp. 737–741. 11, 109

[22] J. C. Church, Y. Chen, and S. V. Rice, "A spatial median filter for noise removal in digital images," in *Southeastcon, 2008 IEEE*. IEEE, 2008, pp. 618–623. 12

[23] J. N. Wilson and G. X. Ritter, *Handbook of computer vision algorithms in image algebra*. CRC press, 2000. 12

[24] D. Charalampidis, "Recursive implementation of the gaussian filter using truncated cosine functions," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3554–3565, 2016. 12

[25] K. Yan, F. Wang, Z. Zhang, N. Li, and F. Lv, "Edge detection of composite insulators hydrophobic image based on improved canny operator," *Energy and Power Engineering*, vol. 5, no. 04, pp. 593–596, 2013. 12

[26] W. K. Pratt, *Digital image processing*. Wiley-Interscience, 2007. 13

[27] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital image processing using MATLAB*. Pearson Education India, 2004. 13, 14, 46, 61, 109

[28] M. Nixon and A. S. Aquado, *Feature extraction and image processing, second edition*. Academic Press, 2008. 13, 41, 109

[29] A. C. Jalba, M. H. Wilkinson, and J. B. Roerdink, "Morphological hat-transform scale spaces and their use in pattern classification," *Pattern Recognition*, vol. 37, no. 5, pp. 901–915, 2004. 13, 60

[30] S. Nomura, K. Yamanaka, T. Shiose, H. Kawakami, and O. Katai, "Morphological preprocessing method to thresholding degraded word images," *Pattern Recognition Letters*, vol. 30, no. 8, pp. 729–744, 2009. 13

[31] Y. Kimori, "Morphological image processing for quantitative shape analysis of biomedical structures: effective contrast enhancement," *Journal of Synchrotron Radiation*, vol. 20, no. 6, pp. 848–853, 2013. 13

[32] G. Wang, Y. Wang, H. Li, X. Chen, H. Lu, Y. Ma, C. Peng, Y. Wang, and L. Tang, "Morphological background detection and illumination normalization of text image with poor lighting," *PLoS ONE*, vol. 9, no. 11, p. e110991, 2014. 13

[33] M. Nieniewski, L. Chmielewski, A. Jozwik, and M. Sklodowski, "Morphological detection and feature-based classification of cracked cegions in ferrites," *Machine Graphics and Vision*, vol. 8, no. 4, pp. 699–712, 1999. 13

[34] A. M. Raid, W. M. Khedr, M. A. El-Dosuky, and M. Aoud, "Image restoration based on morphological operations," *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT)*, vol. 4, no. 3, 2014. 13

[35] D. Rajan and A. Puri, "Segmentation of brain tumour using active contours with level sets," *semanticscholar.org*, pp. 1–25, 2012. 15, 19

[36] C. P. Lee, "Robust image segmentation using active contours: level set approaches," Ph.D. dissertation, Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, North Carolina, 2005. 15, 16, 19

[37] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986. 15

[38] Z. Xianhong and Z. Chunrui, "Canny optimization algorithm based on improved anisotropic diffusion function filtering," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 7, pp. 1–12, 2016. 15

[39] R. Muthukrishnan and M. Radha, "Edge detection techniques for image segmentation," *International Journal of Computer Science and Information Technology*, vol. 3, no. 6, pp. 259–267, 2011. 15

[40] J. A. M. Saif, A. A. M. Al-Kubati, A. S. Hazaa, and M. Al-Moraish, "Image segmentation using edge detection and thresholding," in *The 13th International Arab Conference on Information Technology ACIT'2012*, 2012. 15

[41] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988. 15, 40

[42] C. Solomon and T. Breckon, *Fundamentals of digital image processing: A practical approach with examples in Matlab.* John Wiley and Sons, 2011. 15

[43] M. Hu, X. Ping, and Y. Ding, "Automated cell nucleus segmentation using improved snake," in *International Conference on Image Processing, ICIP'04*, vol. 4. IEEE, 2004, pp. 2737–2740. 15

[44] F. Y. Shih and K. Zhang, "Locating object contours in complex background using improved snakes," *Computer Vision and Image Understanding*, vol. 105, no. 2, pp. 93–98, 2007. 15

[45] T. Guan, D. Zhou, and Y. Liu, "Accurate segmentation of partially overlapping cervical cells based on dynamic sparse contour searching and GVF snake model," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 4, pp. 1494–1504, 2015. 15

[46] V. Caselles, F. Catte, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," *Numerische mathematik*, vol. 66, no. 1, pp. 1–31, 1993. 16, 17, 42

[47] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, vol. 22, no. 1, pp. 61–79, 1997. 16, 109

[48] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations," *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988. 16, 42

[49] J. A. Sethian, "Tracking interfaces with level sets: An act of violence helps solve evolving interface problems in geometry, fluid mechanics, robotic navigation and materials sciences," *American Scientist*, vol. 85, no. 3, pp. 254–263, 1997. 16

[50] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Topology-independent shape modeling scheme," in *SPIE's 1993 International Symposium on Optics, Imaging, and Instrumentation*. International Society for Optics and Photonics, 1993, pp. 246–258. 16, 17, 43

[51] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, "Gradient flows and geometric active contour models," in *proceedings Fifth International Conference on Computer vision*. IEEE, 1995, pp. 810–815. 16

[52] A. Yezzi, S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum, "A geometric snake model for segmentation of medical imagery," *IEEE Transactions on Medical Imaging*, vol. 16, no. 2, pp. 199–209, 1997. 16

[53] H. Shan, C. He, and N. Wang, "MCA aided geodesic active contours for image segmentation with textures," *Pattern Recognition Letters*, vol. 45, pp. 235–243, 2014. 16

[54] S. Shah and A. Ross, "Iris segmentation using geodesic active contours," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 4, pp. 824–836, 2009. 16

[55] K. Bredies and H. Wolinski, "An active-contour based algorithm for the automated segmentation of dense yeast populations on transmission microscopy images," *Computing and Visualization in Science*, vol. 14, no. 7, pp. 341–352, 2011. 16

[56] Z. Zeng, H. Strange, C. Han, and R. Zwiggelaar, "Unsupervised cell nuclei segmentation based on morphology and adaptive active contour modelling," in *International Conference on Image Analysis and Recognition*. Springer, 2013, pp. 605–612. 16

[57] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Evolutionary fronts for topology-independent shape modeling and recovery," in *European Conference on Computer Vision*. Springer, 1994, pp. 1–13. 17, 42

[58] ——, "Shape modeling with front propagation: A level set approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 158–175, 1995. 17, 42

[59] M. B. Salah, A. Mitiche, and I. B. Ayed, "Multiregion image segmentation by parametric kernel graph cuts," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 545–557, 2011. 17

[60] Q. Wu and J. An, "An active contour model based on texture distribution for extracting inhomogeneous insulators from aerial images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 6, pp. 3613–3626, 2014. 17, 19

[61] K. Zhang, H. Song, and L. Zhang, "Active contours driven by local image fitting energy," *Pattern recognition*, vol. 43, no. 4, pp. 1199–1206, 2010. 17

[62] N. Senthilkumaran and S. Vaithegi, "Image segmentation by using thresholding techniques for medical images," *Computer Science and Engineering: An International Journal*, vol. 6, no. 1, 2016. 17

[63] P. D. R. Raju and G. Neelima, "Image segmentation by using histogram thresholding," *International Journal of Computer Science Engineering and Technology*, vol. 2, pp. 776–779, 2012. 17

[64] X. Huang and N. V. Boulgouris, "Robust object segmentation using adaptive thresholding," in *IEEE International Conference on Image Processing, (ICIP 2007)*, vol. 1. IEEE, 2007, pp. I–45. 17

[65] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Aplied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989. 18, 43

[66] Z. L. Szpak and J. R. Tapamo, "Further optimizations for the chan-vese active contour model," in *High Performance Computing and Simulation Conference*. Citeseer, 2008, pp. 272–280. 19

[67] ——, "Maritime surveillance: Tracking ships inside a dynamic background using a fast level-set," *Expert systems with applications*, vol. 38, no. 6, pp. 6669–6680, 2011. 19

[68] W. E. Snyder, "Active contour for multispectral images with non-homogeneous sub-regions," North Carolina State University at Raleigh, Tech. Rep., 2005. 19

[69] M. Lianantonakis and Y. R. Petillot, "Sidescan sonar segmentation using texture descriptors and active contours," *IEEE Journal of Oceanic Engineering*, vol. 32, no. 3, pp. 744–752, 2007. 19

[70] J. Feng, Z. Cao, and Y. Pi, "Multiphase SAR image segmentation with g^0 statistical model based active contours," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 7, pp. 4190–4199, 2013. 19

[71] Q. Wu, J. An, and B. Lin, "A texture segmentation algorithm based on PCA and global minimization active contour model for aerial insulator images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 5, pp. 1509–1518, 2012. 19

[72] D. Frost and J.-R. Tapamo, "Detection and tracking of moving objects in a maritime environment using level set with shape priors," *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, p. 42, 2013. 19

[73] G. Kumar and P. K. Bhatia, "A detailed review of feature extraction in image processing systems," in *Fourth International Conference on Advanced Computing and Communication Technologies (ACCT)*. IEEE, 2014, pp. 5–12. 20

[74] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 3, no. 6, pp. 610–621, 1973. 20, 21

[75] R. M. Haralick, "Statistical and structural approaches to texture," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 786–804, 1979. 20

[76] N. Zayed and H. A. Elnemr, "Statistical analysis of haralick texture features to discriminate lung abnormalities," *Journal of Biomedical Imaging*, vol. 2015, p. 12, 2015. 21

[77] N. Zulpe and V. Pawar, "GLCM textural features for brain tumor classification," *International Jouranl of Computer Science*, vol. 9, no. 3, pp. 354–359, 2012. 21

[78] F. Ahmed, A. H. Bari, A. Shihavuddin, H. A. Al-Mamun, and P. Kwan, "A study on local binary pattern for automated weed classification using template matching and support vector machine," in *IEEE 12th International Symposium on Computational Intelligence and Informatics (CINTI)*. IEEE, 2011, pp. 329–334. 21

[79] D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen, "Local binary patterns and its application to facial image analysis: a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 6, pp. 765–781, 2011. 21, 36

[80] P. Li, X. Lin, J. Jing, and L. Zhang, "Defect detection in fabrics using local binary patterns," in *Chinese Conference on Image and Graphics Technologies*. Springer, 2014, pp. 274–283. 21

[81] F. Tajeripour, E. Kabir, and A. Sheikhi, "Fabric defect detection using modified local binary patterns," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, no. 60, pp. 1–12, 2008. 21

[82] M. Turtinen, M. Pietikainen, and O. Silven, "Visual characterization of paper using isomap and local binary patterns," *IEICE Transactions on Information and Systems*, vol. 89, no. 7, pp. 2076–2083, 2006. 21

[83] B. Yang and S. Chen, "A comparative study on local binary pattern (LBP) based face recognition: LBP histogram versus LBP image," *Neurocomputing*, vol. 120, pp. 365–379, 2013. 21

[84] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996. 22, 64

[85] M. Pietikainen, A. Hadid, G. Zhao, and T. Ahonen, *Local binary patterns for still images*. Springer, 2011. 24

[86] M. Oberweger, A. Wendel, and H. Bischof, "Visual recognition and fault detection for power line insulators," in *19th Computer Vision Winter Workshop*, 2014, pp. 1–8. 25, 32

[87] Z. Zhenbing, L. Ning, and Y. Yapeng, "The recognition and localization of insulators based on SIFT and RANSAC," in *3rd International Conference on Multimedia Technology (ICMT-13)*. Atlantis Press, 2013. 25

[88] R. Azhar, D. Tuwohingide, D. Kamudi, and N. Suciati, "Batik image classification using SIFT feature extraction, bag of features and support vector machine," *Procedia Computer Science*, vol. 72, pp. 24–30, 2015. 25

[89] D. S. Perez, F. Bromberg, and C. A. Diaz, "Image classification of grapevine buds using scale-invariant features transform, bag of features and support vector machines," *arXiv preprint arXiv:1605.02775*, 2016. 25

[90] ——, "Image classification for detection of winter grapevine buds in natural conditions using scale-invariant features transform, bag of features and support vector machines," *Computers and Electronics in Agriculture*, vol. 135, pp. 81–95, 2017. 25

[91] L. Zhang, J. Chen, Y. Lu, and P. Wang, "Face recognition using scale invariant feature transform and support vector machine," in *The 9th International Conference for Young Computer Scientists, (ICYCS 2008)*. IEEE, 2008, pp. 1766–1770. 25

[92] E. Gooya, D. Pastor, and V. Gripon, "Automatic face recognition using SIFT and networks of tagged neural cliques," *Proceedings of Cognitive*, pp. 57–61, 2015. 25

[93] A. Azeem, M. Sharif, J. Shah, and M. Raza, "Hexagonal scale invariant feature transform (H-SIFT) for facial feature extraction," *Journal of Applied Research and Technology*, vol. 13, no. 3, pp. 402–408, 2015. 25

[94] E. Olson, "Particle shape factors and their use in image analysis-part 1: Theory," *Journal of GXP Compliance*, vol. 15, no. 3, pp. 85–96, 2011. 28, 29, 112

[95] ——, "Particle shape factors and their use in image analysis part ii: practical applications," *Journal of GXP Compliance*, vol. 15, no. 4, pp. 77–89, 2011. 28

[96] B. Kaminski, M. Jakubczyk, and P. Szufel, "A framework for sensitivity analysis of decision trees," *Central European Journal of Operations Research*, pp. 1–25, 2017. 29

[97] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wily and Sons, New York, 2000. 29

[98] J. R. Quinlan, "Simplifying decision trees," *International Journal of Man-Machine Studies*, vol. 27, no. 3, pp. 221–234, 1987. 30

[99] E. C. Vasconcellos, R. R. DeCarvalho, R. R. Gal, F. L. LaBarbera, H. V. Capelato, H. F. C. Velho, M. Trevisan, and R. S. R. Ruiz, "Decision tree classifiers for star/galaxy separation," *The Astronomical Journal*, vol. 141, no. 6, pp. 1–12, 2011. 30

[100] E. Akkacs, L. Akin, H. E. Cubukcu, and H. Artuner, "Application of decision tree algorithm for classification and identification of natural minerals using SEM–EDS," *Computers and Geosciences*, vol. 80, pp. 38–48, 2015. 30

[101] H. Deng, G. Runger, and E. Tuv, "Bias of importance measures for multi-valued attributes and solutions," *Artificial Neural Networks and Machine Learning–ICANN 2011*, vol. 6792, pp. 293–300, 2011. 30

[102] E. Fix and J. L. Hodges Jr, "Discriminatory analysis, nonparametric discrimination: consistency properties," Prepared at the University of Califonia,

Berkeley, Defence Technical Information Center (DTIC) Document, Tech. Rep. ADA800276, 1951. 31

[103] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967. 31

[104] S. B. Imandoust and M. Bolandraftar, "Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background," *International Journal of Engineering Research and Applications*, vol. 3, no. 5, pp. 605–610, 2013. 31, 32

[105] D. Masip and J. Vitria, "Shared feature extraction for nearest neighbor face recognition," *IEEE Transactions on Neural Networks*, vol. 19, no. 4, pp. 586–595, 2008. 32

[106] S. Gu, Y. Zheng, and C. Tomasi, "Efficient visual object tracking with online nearest neighbor classifier," *Computer Vision–ACCV 2010*, vol. 6492, pp. 271–282, 2011. 32

[107] M. Kim, Y. Kim, H. Kim, W. Piao, and C. Kim, "Evaluation of the k-nearest neighbor method for forecasting the influent characteristics of wastewater treatment plant," *Frontiers of Environmental Science and Engineering*, vol. 10, no. 2, pp. 299–310, 2016. 32

[108] M. S. Aldayel, "K-nearest neighbor classification for glass identification problem," in *International Conference on Computer Systems and Industrial Informatics (ICCSII)*. IEEE, 2012, pp. 1–5. 32

[109] C. Bo, H. Lu, and D. Wang, "Weighted generalized nearest neighbor for hyperspectral image classification," *IEEE Access*, vol. 5, pp. 1496–1509, 2017. 32

[110] J. Gou, L. Du, Y. Zhang, and T. Xiong, "A new distance-weighted k-nearest neighbor classifier," *Journal of Information and Computational Science*, vol. 9, no. 6, pp. 1429–1436, 2012. 32

[111] S. Taneja, C. Gupta, K. Goyal, and D. Gureja, "An enhanced k-nearest neighbor algorithm using information gain and clustering," in *Fourth International*

Conference on Advanced Computing and Communication Technologies (ACCT). IEEE, 2014, pp. 325–329. 32, 93

[112] B. Sun, J. Du, and T. Gao, "Study on the improvement of k-nearest-neighbor algorithm," in *International Conference on Artificial Intelligence and Computational Intelligence, AICI'09*, vol. 4. IEEE, 2009, pp. 390–393. 32, 93

[113] L. Jiang, Z. Cai, D. Wang, and S. Jiang, "Survey of improving k-nearest-neighbor for classification," in *Fourth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2007*, vol. 1. IEEE, 2007, pp. 679–683. 32, 93

[114] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995. 32, 66

[115] V. S. Murthy, K. Tarakanath, D. K. Mohanta, and S. Gupta, "Insulator condition analysis for overhead distribution lines using combined wavelet support vector machine (SVM)," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 17, no. 1, pp. 89–99, 2010. 32, 36

[116] M. J. B. Reddy, B. K. Chandra, and D. K. Mohanta, "A DOST based approach for the condition monitoring of 11 kv distribution line insulators," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 18, no. 2, pp. 588–595, 2011. 32, 36

[117] M. Reddy and D. K. Mohanta, "A wavelet-support vector machine combined approach for location of transmission line faults," *Australian Journal of Electrical and Electronics Engineering*, vol. 6, no. 1, pp. 45–54, 2009. 32

[118] B. Ravikumar, D. Thukaram, and H. Khincha, "Application of support vector machines for fault diagnosis in power transmission system," *IET Generation, Transmission and Distribution*, vol. 2, no. 1, pp. 119–130, 2008. 32

[119] V. Malathi and N. Marimuthu, "Wavelet transform and support vector machine approach for fault location in power transmission line," *International Journal of Electrical and Electronics Engineering*, vol. 4, no. 4, 2010. 32

[120] P. Ray and D. P. Mishra, "Support vector machine based fault classification and location of a long transmission line," *Engineering Science and Technology, An International Journal*, vol. 19, no. 3, pp. 1368–1380, 2016. 32

[121] K. S. Parikh and T. P. Shah, "Support vector machine–a large margin classifier to diagnose skin illnesses," *Procedia Technology*, vol. 23, pp. 369–375, 2016. 32, 67

[122] Y. Libo and C. Hao, "Face recognition based on the combination method of multiple classifier," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 4, pp. 151–164, 2016. 32

[123] M. Abdulrahman and A. Eleyan, "Facial expression recognition using support vector machines," in *23th Signal Processing and Communications Applications Conference (SIU)*. IEEE, 2015, pp. 276–279. 32

[124] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," *Department of Computer Science, National Taiwan University*, pp. 1–16, 2003. 34

[125] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 1999, pp. 42–49. 34

[126] I. Y. Gu, U. Sistiaga, S. M. Berlijn, and A. Fahlstrom, "Online detection of snow-coverage and swing angles of electrical insulators on power transmission lines using videos," in *16th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2009, pp. 3249–3252. 35

[127] X. Mei, T. Lu, X. Wu, and B. Zhang, "Insulator surface dirt image detection technology based on improved watershed algorithm," in *Asia-Pacific Power and Energy Engineering Conference (APPEEC)*. IEEE, 2012, pp. 1–5. 35

[128] Y. Ge, B. Li, S. Zhao, and C. Pang, "Detection of the insulator dirtiness based on the computer vision," in *China International Conference on Electricity Distribution (CICED 2006)*. IET, 2006, pp. 173–176. 35

[129] W. Li, G. Ye, F. Huang, S. Wang, and W. Chang, "Recognition of insulator based on developed mpeg-7 texture feature," in *3rd International Congress on Image and Signal Processing (CISP)*, vol. 1. IEEE, 2010, pp. 265–268. 35

[130] J. Zhang and R. Yang, "Insulators recognition for 220kv/330kv high-voltage live-line cleaning robot," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 4. IEEE, 2006, pp. 630–633. 35

[131] V. S. Murthy, S. Gupta, and D. K. Mohanta, "Digital image processing approach using combined wavelet hidden markov model for well-being analysis of insulators," *IET Image Processing*, vol. 5, no. 2, pp. 171–183, 2011. 36

[132] G. Zhao, T. Ahonen, J. Matas, and M. Pietikainen, "Rotation-invariant image and video description with local binary pattern features," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 165–1477, 2012. 36

[133] M. Pietikainen, "Local binary patterns," *Scholarpedia, 5(3):9775*, 2010. 36

[134] M. Berg, R. Thottappillil, and V. Scuka, "Hydrophobicity estimation of HV polymeric insulating materials. development of a digital image processing method," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 8, no. 6, pp. 1098–1107, 2001. 37, 105

[135] C. Liang, W. Yang, and Q. Liao, "Water droplets segmentation for hydrophobicity classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 1181–1184. 37, 115, 116, 121

[136] X. Chen, C. Li, X. Huang, L. Zhao, and W. Song, "On-line estimating the level of hydrophobicity of composite insulators using the digital images," in *Electrical Insulation Conference and Electrical Manufacturing Expo, 2005. Proceedings.* IEEE, 2005, pp. 216–221. 37

[137] T. Tokoro, M. Nagao, and M. Kosaki, "Image analysis of hydrophobicity of silicone rubber insulator," in *Annual Report Conference on Electrical Insulation and Dielectric Phenomena*, vol. 2. IEEE, 1999, pp. 763–766. 37

[138] C. Li, X. Huang, and L. Zhao, "Image analysis on the surface hydrophobicity of polluted silicone rubber insulators," in *International Conference on Condition Monitoring and Diagnosis (CMD2008)*. IEEE, 2008, pp. 389–391. 37, 107

[139] Y. Qiuxia, T. Liangrui, Q. Bing, and Z. Jing, "A new fcm-based algorithm of hydrophobic image segmentation," in *Sixth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD'09*, vol. 3. IEEE, 2009, pp. 374–377. 37

[140] D. Thomazini, M. V. Gelfuso, and R. A. C. Altafim, "Hydrophobicity classification of polymeric materials based on fractal dimension," *Materials Research*, vol. 11, no. 4, pp. 415–419, 2008. 37

[141] I. Jarrar, K. Assaleh, and A. H. El-Hag, "Using a pattern recognition-based technique to assess the hydrophobicity class of silicone rubber materials," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 21, no. 6, pp. 2611–2618, 2014. 37

[142] L. Khalayli, H. Al Sagban, H. Shoman, K. Assaleh, and A. Ei-Hag, "Automatic inspection of outdoor insulators using image processing and intelligent techniques," in *Electrical Insulation Conference (EIC), 2013 IEEE*. IEEE, 2013, pp. 206–209. 38

[143] S. Goel and S. K. Mathew, "Automated detection, characterization, and tracking of sunspots from SoHO/MDI continuum images," *Solar Physics*, vol. 289, no. 4, pp. 1413–1431, 2014. 39

[144] P. G. Scholor and S. P. S. Subramanian, "Extraction and classification of blebs in human embryonic stem cell," *International e-Journal For Technology And Research*, vol. 1, no. 2, pp. 1–8, 2017. 40, 41

[145] U. Iruansi, J. R. Tapamo, I. E. Davidson, and M. Khan, "Insulator region of interest using active contour model and speeded up robust features," in *Proceedings of the 2nd Eskom Power Plant Engineering Institute (EPPEI) Student Workshop, Eskom Academy of Learning*, 2015, pp. 52–53. 46, 47

[146] U. Iruansi, J. R. Tapamo, and I. E. Davidson, "An active contour approach to insulator segmentation," in *Proceeding of the 12th IEEE AFRICON International Conference*. IEEE, 2015, pp. 1–5. 47, 58

[147] "Pearson education inc., table a-3, t-distribution: critical t-values," http://www.math.odu.edu/stat130/t-tables.pdf, Accessed: March 5, 2017. 56

[148] J. H. Min and Y.-C. Lee, "Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters," *Expert Systems with Applications*, vol. 28, no. 4, pp. 603–614, 2005. 67, 68

[149] D. M. W. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *Journal of Machine Learning*, vol. 2, no. 1, pp. 37–63, 2011. 75, 115

[150] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006. 75, 115

[151] F. Albregtsen, "Statistical texture measures computed from gray level coocurrence matrices," *Image Processing Laboratory, Department of Informatics, University of Olso*, vol. 5, pp. 1–14, 2008. 77

[152] M. V. Boland, "Haralick texture features," http://murphylab.web.cmu.edu/publications/boland/boland_node26.html, Online; accessed June 6, 2017. 77

[153] O. A. Adedokun and W. D. Burgess, "Analysis of paired dichotomous data: a gentle introduction to the mcnemar test in SPSS," *Journal of MultiDisciplinary Evaluation*, vol. 8, no. 17, pp. 125–131, 2011. 82, 83, 101

[154] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998. 82, 83, 101

[155] "Chi-square distribution table," http://sites.stat.psu.edu/~mga/401/tables/Chi-square-table.pdf, Accessed: April 12, 2017. 83, 102

[156] P. Vinukonda, "A study of the scale-invariant feature transform on a parallel pipeline," Master's thesis, Louisiana State University, 2011. 90

[157] M. Amin, M. Akbar, and S. Amin, "Hydrophobicity of silicone rubber used for outdoor insulation (an overview)," *Reviews on Advanced Material Science*, vol. 16, pp. 10–26, 2007. 105

[158] R. Sundararajan, A. Mohammed, N. Chaipanit, T. Karcher, and Z. Liu, "In-service aging and degradation of 345kV EPDM transmission line insulators in a coastal environment," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 11, no. 2, pp. 348–361, 2004. 105

[159] K. Naito, R. Matsuoka, T. Irie, and K. Kondo, "Test methods and results for recent outdoor insulation in japan," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 6, no. 5, pp. 732–743, 1999. 105

[160] U. Iruansi, J. R. Tapamo, I. E. Davidson, and M. Khan, "Polymeric insulator condition analysis based on hydrophobicity," in *Proceedings of the 3rd Eskom Power Plant Engineering Institute (EPPEI) Student Workshop, Eskom Academy of Learning*, 2016, pp. 44–45. 109

[161] W. Burger and M. J. Burge, *Principles of digital image processing.* Springer, 2009. 111